NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

ASYNCHRONOUS TRANSFER MODE AND LOCAL AREA NETWORK EMULATION STANDARDS, PROTOCOLS, AND SECURITY IMPLICATIONS

by

John P. Kirwin

December 1999

Thesis Advisor:	John McEachen
Second Reader:	Murali Tummala

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 1999	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE **ASYNCHRONOUS TRANSFER MODE AND LOCAL AREA NETWORK EMULATION STANDARDS, PROTOCOLS, AND SECURITY IMPLICATIONS**	5. FUNDING NUMBERS
6. AUTHOR(S) Kirwin, John P.	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8.PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Naval Information Warfare Activity 9800 Savage Road, Ft. Meade, MD 20755	10.SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited.	12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

A complex networking technology called Asynchronous Transfer Mode (ATM) and a networking protocol called Local Area Network Emulation (LANE) are being integrated into many naval networks without any security-driven naval configuration guidelines. No single publication is available that describes security issues of data delivery and signaling relating to the transition of Ethernet to LANE and ATM. The thesis' focus is to provide: (1) an overview and security analysis of standardized protocols relating to ATM and LANE; (2) an overview and security analysis associated with integrating a Fore Systems Inc., LANE-based ATM network, with an accredited Cisco Systems Inc., Ethernet Virtual LAN (VLAN) network; and (3) associated security-related suggestions for network design and configurations. This thesis identifies possible negative security-related capabilities associated with ATM- and LANE-related protocols; however, many can be mitigated using the identified network design guidelines. Qualitative analysis suggests that the introduction of an ATM/LANE backbone into an existing TCP/IP network does not increase the probability of incorrect destinations receiving and processing corrupted frames. It is hoped that this seminal document will assist in the development of standard security-driven implementation guidelines associated with ATM/LANE-based networks, as well as inform those required to prepare and review associated network Risk Assessments.

14. SUBJECT TERMS KEYWORDS: Asynchronous Transfer Mode, ATM LAN Emulation, LANE, Emulated LAN, ELAN, Security, Private Network Network Interface, PNNI, User Network Interface, UNI			15. NUMBER OF PAGES 149
			16. PRICE CODE
17.SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFI- CATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

THIS PAGE INTENTIONALLY LEFT BLANK

ASYNCHRONOUS TRANSFER MODE AND LOCAL AREA NETWORK EMULATION STANDARDS, PROTOCOLS, AND SECURITY IMPLICATIONS

John P. Kirwin
Civilian
B.S.E.E., Pennsylvania State University, 1989

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1999

Author: ______________________________
John P. Kirwin

Approved by: ______________________________
John McEachen, Thesis Advisor

____
Murali Tummala, Second Reader

____
Jeffrey B. Knorr, Chairman
Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

A complex networking technology called Asynchronous Transfer Mode (ATM) and a networking protocol called Local Area Network Emulation (LANE) are being integrated into many naval networks without any security-driven naval configuration guidelines. No single publication is available that describes security issues of data delivery and signaling relating to the transition of Ethernet to LANE and ATM. The thesis' focus is to provide: (1) an overview and security analysis of standardized protocols relating to ATM and LANE; (2) an overview and security analysis associated with integrating a Fore Systems Inc., LANE-based ATM network, with an accredited Cisco Systems Inc., Ethernet Virtual LAN (VLAN) network; and (3) associated security-related suggestions for network design and configurations. This thesis identifies possible negative security-related capabilities associated with ATM- and LANE-related protocols; however, many can be mitigated using the identified network design guidelines. Qualitative analysis suggests that the introduction of an ATM/LANE backbone into an existing TCP/IP network does not increase the probability of incorrect destinations receiving and processing corrupted frames. It is hoped that this seminal document will assist in the development of standard security-driven implementation guidelines associated with ATM/LANE-based networks, as well as inform those required to prepare, and review associated network Risk Assessments.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS ACRONYM'S AND SYMBOLS

1.	AAL	Asynchronous Adaptation Layer
2.	AAL5	ATM Adaptation Layer 5
3.	AED	ATM Edge Device
4.	AES	ATM End Station
5.	AESA	ATM End System Address
6.	AFI	Authority Format Identifier
7.	AIS	Alarm Indication Signal
8.	AIS-L	Alarm Indication Signal – Line
9.	AIS-P	Alarm Indication Signal – Path
10.	ANSI	American National Standards Institute
11.	ARP	Address Resolution Protocol
12.	AS	ATM Switch
13.	ASN.1	Abstract Syntax Notation
14.	ATM	Asynchronous Transfer Mode
15.	AToM MIB	Definitions of Managed Objects for ATM Management
16.	B	Byte
17.	BIP	Bit Interleaved Parity
18.	BIP8	Bit Interleaved Parity – 8 bits
19.	B-ISDN	Broadband – Integrated Services Digital Network
20.	BLLI	Broadband Low-Layer Informaiton
21.	BPDU	Bridge Protocol Data Unit
22.	BUS	Broadcast Unknown Server
23.	CA	Cache Alignment
24.	CAC	Connection Admission Control
25.	CACHESYN	Cache Synchronous
26.	CDP	Cisco Discovery Protocol
27.	CLI	Command Line Interface
28.	CLP	Cell Loss Priority
29.	CONFDIR	Configuration Direct
30.	CONTDIR	Control Direct
31.	CONTDIS	Control Distribute
32.	CPCS	Common Part ConvergenceSublayer
33.	CPCS-UU	CPCS- User –to-User
34.	CPG	Child Peer Group
35.	CPI	Common Part Indicator

36. C-Plane	Control Plane
37. CRC	Cyclic Redundancy Check
38. CRC-32	Cyclic Redundancy Check – 32 bits
39. CRL	Cache State Alignment Request List
40. CSA	Cache State Alignment
41. CSAS	Cache State Alignment Summary
42. CSU	Cache State Update
43. CSUS	Cache State Update Solicit
44. DATDIR	Data Direct
45. DCS	Directly Connected Station
46. DEFMULSND	Default Multicast Send
47. DESN	Destination Sequence Number
48. DISA	Defense Information Systems Agency
49. DIX	Digital-Intel-Xerox
50. DLE	Distributed LAN Emulation
51. DNS	Domain Name Server
52. DS	Digital Signal Hierarchy
53. DS1/T1	Digital Signal Hierarchy, Level 1 / T-carrier Service 1
54. DS3	Digital Signal Hierarchy, Level 3
55. DSAP	Destination Service Access Point
56. DTL	Designated Transit List
57. E1	European Transmission Service, Level 1
58. EES	Ethernet End Station
59. ELAN	Emulated LAN
60. ER	Error Recovery
61. ESI	End System Identifier
62. FEBE	Far End Block Error
63. FEBE-L	Far End Block Error - Line
64. FEBE-P	Far End Block Error - Path
65. FSSRP	Fast Simple Server Redundancy Protocol
66. FTP	File Transfer Protocol
67. GFC	Generic Flow Control
68. HEC	Header Error Check
69. IANA	International Assigned Numbers Authority
70. ICD	International Code Designator
71. ICMP	Internet Control Message Protocol
72. IE	Information Element

73. IEEE	Institute of Electrical and Electronic Engineers
74. IETF	Internet Engineering Task Force
75. IGMP	Internet Group Management Protocol
76. ILMI	Interim Local Management Interface
77. INFOSEC	Information Security
78. IP	Internet Protocol
79. ISDN	Integrated Services Digital Network
80. ISL	Inter Switch Link
81. ISO/IEC	International Organization for Standardization/International Electrotechnical Commission
82. ITU-T	International Telecommunications Union-Telecommunications
83. LANE	Local Area Network Emulation
84. LANE v1	Local Area Network Emulation version 1
85. LANE v2	Local Area Network Emulation version 2
86. LEC	LANE Client
87. LECMv2	LEC Management Specification version 2
88. LECSSYN	LECS Synchronization
89. LES	LAN Emulation Server
90. LESMv1	LES Management Specification version 1
91. LGN	Logical Group Node
92. LLC	Logical Link Control
93. LNNI v2	LANE Network-Network Interface
94. LOF	Loss of Frame
95. LOH	Line Overhead
96. LOS	Loss of Signal
97. LPN	Leadership Priority Number
98. LS	Local Station
99. LSID	Local Station Identification
100.LTE	Line Terminating Equipment
101.LUNI v2	LANE User Network Interface version 2
102.MAC	Media Access Control
103.MAIB	Message Action Indicator Bits
104.MIB-II	Management Information Base II
105.MIE	Message Information Element
106.M-Plane	Management Plane
107.MPOA	MultiProtocol Over ATM
108.MSAB	Most Significant Address Bits

109.MULDIR	Multicast Direct
110.MULFOR	Multicast Forward
111.NAM	Network Analysis Module
112.NDF	New Data Flag
113.NetX	Network X
114.NRM	Network Resource Management
115.OAM	Operations Administration and Maintenance
116.OCD	Out of Cell Delineation
117.OC-N	Optical Carrier -N
118.PAD	Pad
119.PDU	Protocol Data Unit
120.PG	Peer Group
121.PGID	Peer Group Identification
122.PGL	Peer Group Leader
123.PICS	Protocol Implementation Conformance Statement
124.PLCP	Physical Layer Convergence Protocol
125.PM	Physical Medium
126.PNNI	Private Network to Network Interface
127.POH	Path Overhead
128.PPG	Parent Peer Group
129.PT	Payload Type
130.PTE	Path Terminating Equipment
131.PTSE	PNNI Topology State Element
132.PTSE-ACK	PNNI Topology State Element - Acknowledgment
133.PTSP	PNNI Topology State Packet
134.PVC	Permanent Virtual Circuit
135.QOS	Quality of Service
136.RARP	Reverse Address Resolution Protocol
137.RCC	Routing Control Channel
138.RCS	Remotely Connected Station
139.RDI	Remote Defect Indication
140.RDI-L	Remote Defect Indication - Line
141.RDI-P	Remote Defect Indication - Path
142.RFC	Request For Comment
143.RH	Rogue Host
144.SAAL	Signaling AAL
145.SAR	Segmentation and Reassembly

146.SCD	Selective Cell Discarding
147.SCSP	Server Cache Synchronous Protocol
148.SDH	Synchronous Digital Hierarchy
149.SDU	Service Data Unit
150.SEF	Severely Errored Frame
151.SEL	Selector
152.SELMULSND	Selective Multicast Send
153.SID	Server Identification
154.SMDS	Switched Multimegabit Data Services
155.SMS	Selective Multicast Server
156.SNMP	Simple Network Management Protocol
157.SNMP v2	Simple Network Management Protocol
158.SOH	Section Overhead
159.SONET	Synchronous Optical Network
160.SPANS	Simple Protocol for ATM Network Signaling
161.SPE	Synchronous Payload Envelope
162.SPVC	Soft Permanent Virtual Circuit
163.SSAP	Source Service Access Point
164.SSCF	Service Specific Coordination Function
165.SSCOP	Service Specific Connection Oriented Protocol
166.SSCS	Service Specific Convergence Sublayer
167.SSN	Synchronize Sequence Number
168.STE	Section Terminating Equipment
169.STS	Synchronous Transport Signal
170.SVC	Switched Virtual Circuit
171.TC	Transmission Convergence
172.TCP	Transmission Control Protocol
173.TH	Target Host
174.TLV	Type/Length/Value
175.TOH	Transport Overhead
176.UDP	User Datagram Protocol
177.ULIA	Uplink Information Attribute
178.UME	UNI Management Entity
179.UNI	User Network Interface
180.UPC	Usage Parameter Control
181.U-Plane	User Plane
182.USTAT	Unsolicited Status

183.VBR	Variable Bit Rate
184.VC	Virtual Circuit
185.VC-AIS	Virtual Container – Alarm Indication Signal
186.VCC	Virtual Channel Connections
187.VCD	Virtual Circuit Descriptor
188.VCI	Virtual Channel Identifier
189.VCL	Virtual Circuit Link
190.VC-RDI	Virtual Container Remote Defect Indicator
191.VMPS	VLAN Membership Policy Server
192.VP	Virtual Path
193.VP-AIS	Virtual Path – Alarm Indication Signal
194.VPC	Virtual Path Connection
195.VPI	Virtual Path Identifier
196.VPL	Virtual Path Link
197.VP-RDI	Virtual Path – Remote Defect Indicator
198.VPT	Virtual Path Terminator
199.VTP	Virtual Through Path
200.WAN	Wide Area Network

ACKNOWLEDGMENT

I would like to take this opportunity to thank the individuals who supported me through my research and writing of this thesis. To Professor John McEachen for his support and willingness to be my thesis advisor, as well as Professor Murali Tummala for being my second reader and providing my first course in Broadband Networks. I would also like to thank the Naval Surface Warfare Center – Dahlgren Division, especially my supervisor Rodney Todt for his support and patience. To Fore Systems Inc., technical support. To wife Jean, my son John, and my family whom without their support, prayers and inspiration I would not have been able to complete this.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Asynchronous Transfer Mode (ATM) is one of the most versatile, but complex networking technologies conceived to date. It can transport voice, video, or data by offering various types of Qualities of Service (QOS) in terms of latency, timing, and transfer rate. It is scalable in that the speed and size of a network can grow to accommodate bandwidth expansion without revamping an entire network. Another attractive aspect of ATM is that it can transport preexisting, and future communication technologies transparently: such as leased telephone lines (e.g. DS1/T1, E1, DS3), Integrated Services Digital Network (ISDN), Ethernet (Digital-Intel-Xerox DIX and IEEE 802.3), Token Ring, and Fiber Distributed Data Interface (FDDI)). With technology moving as fast as it is today, ATM is the only technology that has matured enough to offer the diversity, scalability, and transparency to meet projected communications needs well into the foreseeable future.

Despite ATM's advantages and maturity, it is still a relatively new networking technology that is being used by many naval bases without any standard procedures defined for its configuration and use. There are literally thousands of pages associated with the protocols associated with ATM. The number and complexity of the protocols is such that an in-depth security analysis would be costly to contract or undertake by an organization. Further, no one book, paper, or journal is available that describes security issues of data delivery and signaling relating to the transition of Ethernet to Local Area Network Emulation (LANE) and ATM. A summarized description of the operation and security issues of each protocol associated with LANE and ATM is needed so that informed trade-off studies and security measures can be taken to assist naval ATM network security accreditors.

The accomplishments of this study are three-fold. First it provides a description of the operation and security issues of standardized ATM and LANE related protocols. Second it provides an overview and identifies security issues of integrating a LANE based ATM Fore Systems Inc., backbone network, into an accredited Cisco Systems Inc., based Ethernet Virtual LAN (VLAN) network. Third it suggests network design and configuration settings to Fore and Cisco based equipment to mitigate security vulnerabilities noted.

In an effort to clarify content, an Ethernet and ATM network model, referred to as "NetX" Figure 1, is provided, as well as a general overview of TCP/IP related protocols. NetX is assumed to be customer owned and in a controlled facility.

The use of an accredited Cisco Ethernet VLAN network abrogates the need to incorporate a security analysis of Ethernet routers or Cisco's proprietary VLAN technology within this document. An overview and suggestions are made to Cisco VLAN configuration in an effort to safeguard IP access to Fore and Cisco console ports as well as to better secure a network in general.

Transmission error outcome and possible protocol and vendor related security vulnerabilities sited in this document are the result of an analysis of the relevant standards and publicly available vendor product literature. This study does not reflect actual vendor specific implementations, expose verified vulnerabilities, nor means to exploit these vulnerabilities.

It is hoped that this seminal document will assist in the development of standard security-driven implementation and operation procedures associated with ATM/LANE based networks, as well as to inform those employees required to prepare and review associated network Risk Assessments.

A. A BRIEF HISTORY OF ATM AND SECURITY GUIDELINES

ATM began in early 80's with the adoption of Narrow-band Integrated Services Digital Network (ISDN). In 1986, the International Telecommunications Union – Telecommunications (ITU-T), chose the ATM approach for the Broadband - Integrated Services Digital Network (B-ISDN). By June of 1989, an international standard ATM cell size (53 bytes) was defined, and in 1991, the ATM Forum was founded. Since then, core supporting protocols such as the User Network Interface (UNI), and Private Network to Network Interface (PNNI) have been developed, as well as a host of others. Contributing members not only include the ATM Forum and ITU-T, but also the American National Standard Institute (ANSI), and the Internet Engineering Task Force (IETF).

The ATM forum first addressed the need for interoperable security for ATM in early 1995 in a number of contributions that addressed high level requirements. A clear need for standardization of ATM was articulated; consequently, the Security Working Group was officially formed in October 1995. [1]

In September 1997, Bellcore completed an ATM Network Security Assessment for the Defense Information Systems Agency (DISA) – D383. The purpose and intent of this document was to provide DISA personnel with an overview of ATM networks, an explanation of security vulnerabilities, potential threats, requirements, recommendations for security services, and mechanisms to mitigate related threats and vulnerabilities.

In February 1998, the ATM Forum Technical Committee created the ATM Security Framework Version 1.0. This framework identified the fundamental generic security objectives that must be met with a security specification, the threats, security requirements, and services, as well as an approach to solving security problems based upon the ATM model. The framework was a valuable starting point, however, it did not map any security services to the ATM network, architecture, user plane, control plane, and management planes, nor did it identify a mechanism, nor algorithms to realize the security services within the Security Framework.

One year after the release of the Security Framework, the ATM Forum released the ATM Security Specification Version 1.0. This specification provided security services for the user, and control plane, and provided other supporting services. The Security Specification does not directly support the management plane; however, it does indirectly support some of the management plane security services via supporting user plane security services[2]. In-band security message exchange services are provided for the negotiation of encryption services, as well as the transfer of encryption related parameters. In May 1999, ATM Forum released the UNI 4.0, and PNNI 1.0 Signaling Security addenda. This addendum provided a means for an ATM network to use the ATM Security Specification services.

According to the Department of Navy Information Systems Security (INFOSEC) [INFOSEC work order 9900272], there are no known standard naval guidelines specifically related to safeguarding against ATM security vulnerabilities.

B. A BASIC NETWORK MODEL CALLED NETWORK X (NETX)

This thesis focuses on the introduction of an ATM backbone network into a preexisting accredited Ethernet VLAN network. The combined Ethernet/ATM network model called "NetX" is shown in

Figure 1. NetX represents an example of the connectivity between six major physical elements of a network to provide Ethernet and ATM connectivity. The six major elements of NetX are:

1. Ethernet End Stations (EES) - An EES is a Personal Computer (PC), that uses a TCP/IP protocol suite of software and an IEEE 802.3 Ethernet card to communicate with other Ethernet users.

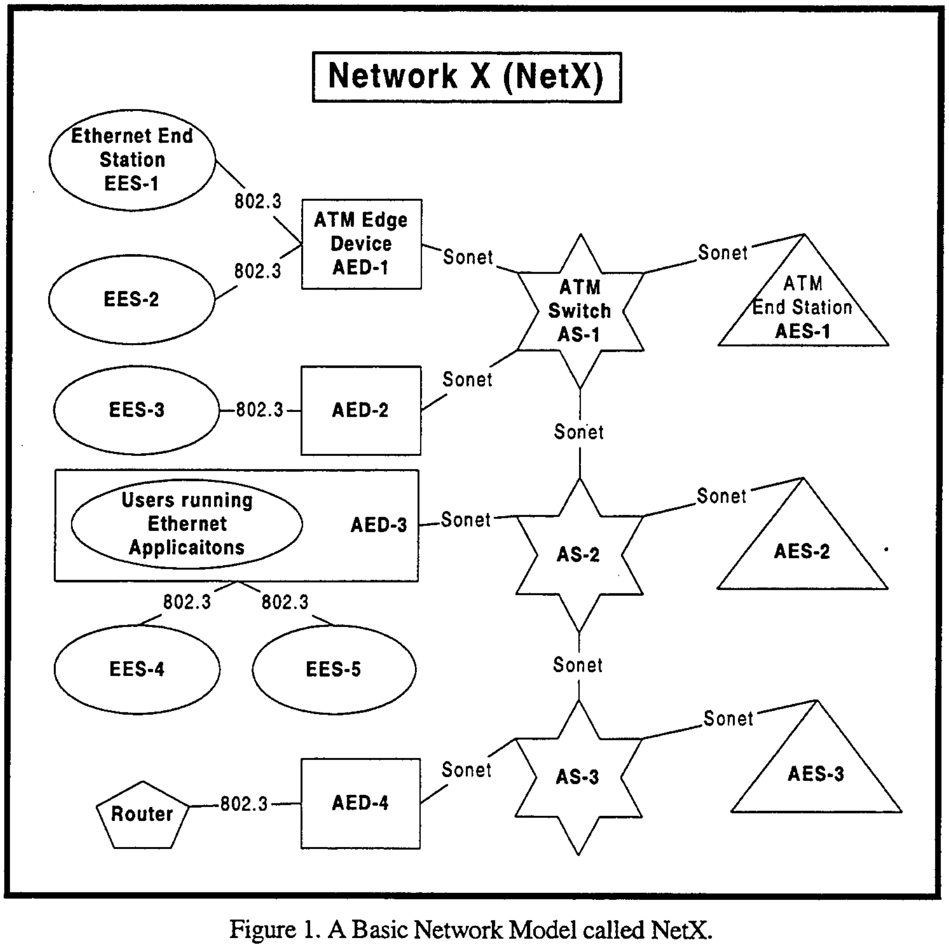


Figure 1. A Basic Network Model called NetX.

2. ATM Edge Devices (AED) - An AED is a device that is capable of forwarding information between legacy interfaces (e.g., Ethernet, Token Ring) and ATM networks based on the data link and network layer information, but does not participate in running network layer routing protocols.
3. ATM Switches (AS) - An AS is a device that participates in PNNI routing, and PNNI and/or UNI signaling to enable communication between ATM devices.
4. ATM End Stations (AES) - An AES is a device that is directly connected to an AS, but does not forward information between legacy networks and ATM networks.
5. Router - A Router is a device that participates in the forwarding of packets based on network layer information and participates with one or more routing protocols. Routers interconnect LANs as well as WANs. A router and an AED can both be incorporated into a single physical device.
6. Physical Links - A physical link is a means by which two devices physically connect. In the context of this paper, all ATM interfaces physically interconnect via the Synchronous Optical Network (SONET) standard using fiber optic cable, and all Ethernet devices physically interconnect via the IEEE 802.3 standard using either electrical cable or fiber optic cable.

NetX permits Ethernet based communication between devices that have Ethernet applications. An Ethernet application such as the Address Resolution Protocol (ARP), may run on an Ethernet End Station (EES), or ATM Edge Device (AED) running LAN Emulation (LANE) software. NetX also permits ATM based communication between devices that have ATM applications. ATM applications, such as LANE, run on ATM devices such as an ATM Switch (AS), AED, or ATM Enc Station (AES).

C. SECURITY GOALS AND TERMS

A secure network should inherently provide a security service to insure the confidentiality, and data integrity of stored, and transferred information. A secure network should also provide some measure of guarantee that the network will be accessible, and that all elements are accountable for the actions that they perform.

A threat may be accidental, or intentional. Possible sources of threats could arise from software, or hardware failures, or flaws in their design and/or configuration. A threat may also present itself from a lack of proper security implementation, procedures, and/or administration. Threats that should be included in a security analysis are identified and defined by the ATM Forum [3] below:

1. **Masquerade ("spoofing")** – The pretense by an entity to be a different entity.
2. **Eavesdropping** - A breach of confidentiality by monitoring communication.
3. **Unauthorized access** - An entity attempts to access data in violation of the security policy in force.
4. **Loss or corruption of information** - The integrity of data transferred is compromised by unauthorized: deletion, insertion, modification, reordering, replay, or delay.
5. **Repudiation** - An entity involved in a communication exchange subsequently denies the fact.
6. **Forgery** - An entity fabricates information and claims that such information was received from another entity, or sent to another entity.
7. **Denial of Service** - This occurs when an entity fails to perform its function or prevents other entities from performing their functions. This threat may include denial of access to ATM services, and denial of communication by flooding an ATM network/component. In a shared network, this

threat can be recognized as a fabrication of extra traffic that floods the network, preventing others from using the network or delaying the traffic of others.

Understanding how a network works, is a precursor to understanding how to protect it. This document first describes many of the protocols associated with NetX, (Chapters II - IV) then points out positive and negative security points associated with each. This analysis of the protocols provides an overview of negative security-related capabilities (Chapter VI). Recall, the assumption thus far has been the network does not have specialized protection mechanisms other than that provided by the protocols themselves. A means for providing additional security to a network (without the use of firewalls) using physical security and software included with Fore System Switches and Cisco Catalyst switches is then provided (Chapter VIII). Chapter IX provides a high level conclusion to this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

II. THE TCP/IP/802.3 PROTOCOL SUITE

The TCP/IP protocol suite in this thesis refers to a number of protocols including: Address Resolution Protocol (ARP), Reverse Address Resolution Protocol (RARP), Internet Control Message Protocol (ICMP), Internet Group Message Protocol (IGMP), Internet Protocol (IP), Transmission Control Protocol (TCP), and User Datagram Protocol (UDP). Examples of applications that use this protocol suite are the Simple Network Management Protocol (SNMP), and File Transfer Protocol (FTP).

A. THE TCP/IP PROTOCOL SUITE

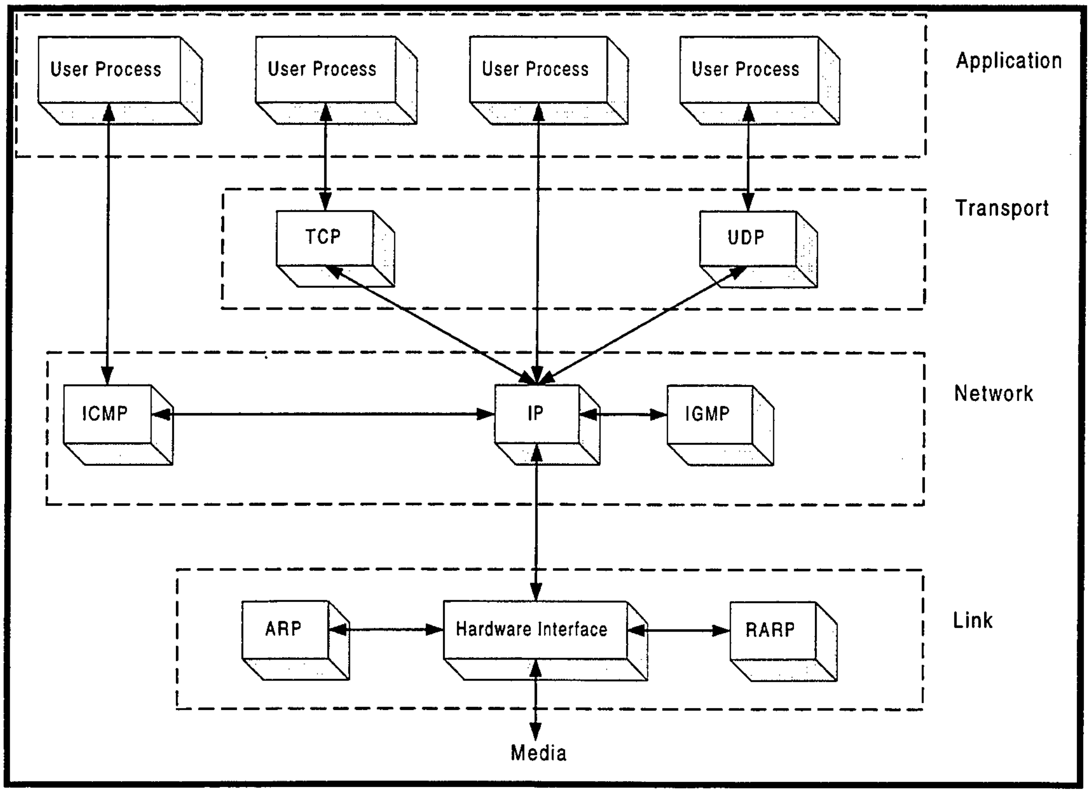


Figure 2. A High-level View of the TCP/IP Protocol Suite

NetX Ethernet user applications on an EES, or AED typically use the Internet Control Message Protocol (ICMP), Transmission Control Protocol (TCP), or User Datagram Protocol (UDP) to communicate over a network. Figure 2 illustrates the relationships between user applications, network protocols, and layers. Figure 3 (a detailed view of Figure 2) illustrates how sensitive user application information is packaged, or unpackaged as it moves through hierarchical protocol layers.

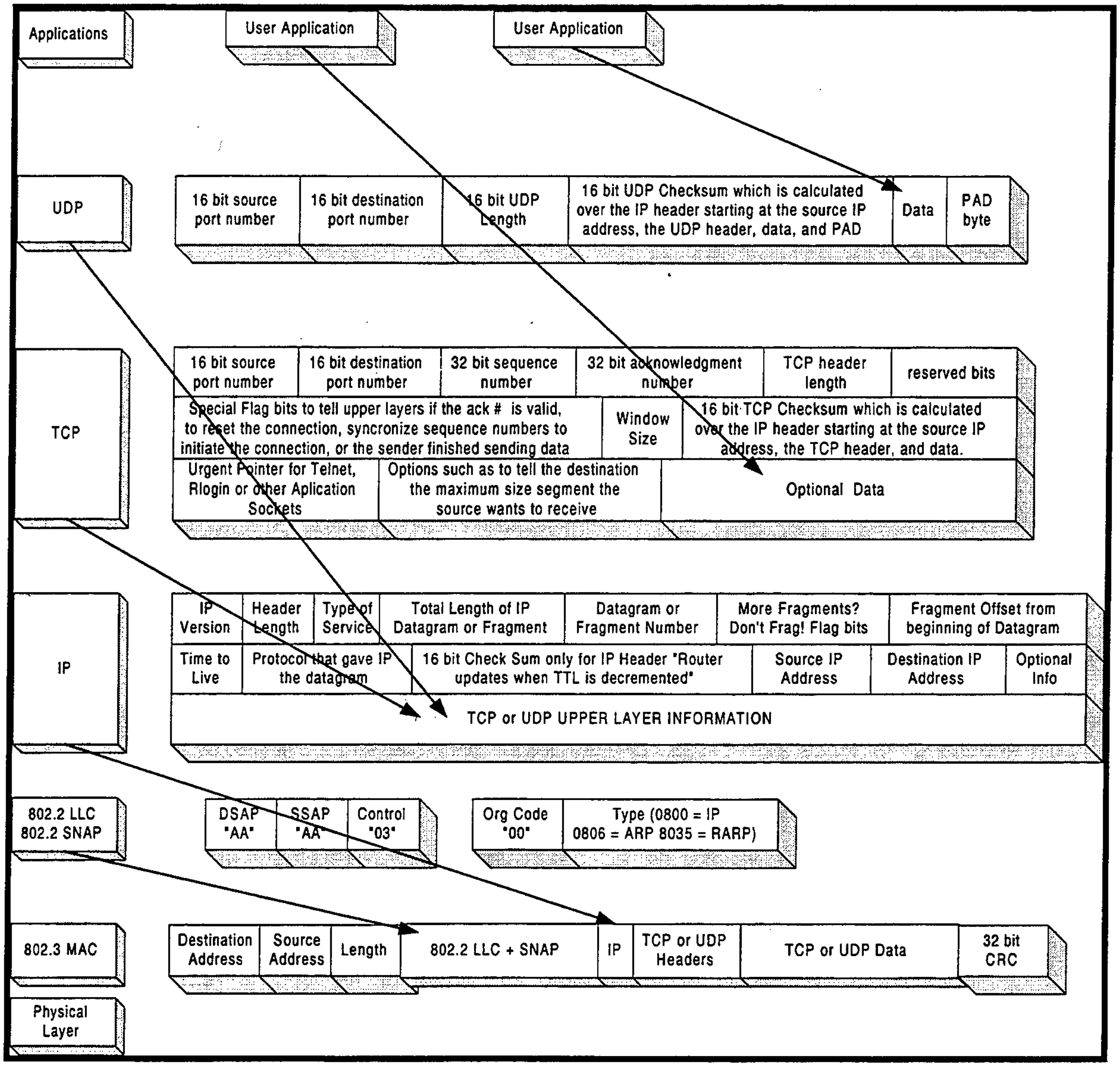

Applications
User Application
User Application
UDP
16 bit source port number
16 bit destination port number
16 bit UDP Length
16 bit UDP Checksum which is calculated over the IP header starting at the source IP address, the UDP header, data, and PAD
Data
PAD byte
TCP
16 bit source port number
16 bit destination port number
32 bit sequence number
32 bit acknowledgment number
TCP header length
reserved bits
Special Flag bits to tell upper layers if the ack # is valid, to reset the connection, syncronize sequence numbers to initiate the connection, or the sender finished sending data
Window Size
16 bit TCP Checksum which is calculated over the IP header starting at the source IP address, the TCP header, and data.
Urgent Pointer for Telnet, Rlogin or other Aplication Sockets
Options such as to tell the destination the maximum size segment the source wants to receive
Optional Data
IP
IP Version
Header Length
Type of Service
Total Length of IP Datagram or Fragment
Datagram or Fragment Number
More Fragments? Don't Frag! Flag bits
Fragment Offset from beginning of Datagram
Time to Live
Protocol that gave IP the datagram
16 bit Check Sum only for IP Header "Router updates when TTL is decremented"
Source IP Address
Destination IP Address
Optional Info
TCP or UDP UPPER LAYER INFORMATION
802.2 LLC 802.2 SNAP
DSAP "AA"
SSAP "AA"
Control "03"
Org Code "00"
Type (0800 = IP 0806 = ARP 8035 = RARP)
802.3 MAC
Destination Address
Source Address
Length
802.2 LLC + SNAP
IP
TCP or UDP Headers
TCP or UDP Data
32 bit CRC
Physical Layer

Figure 3. A Detailed View of the TCP/IP Protocol Suite

Examples of applications that use ICMP are "traceroute" and "ping." The ICMP protocol is used to communicate error messages or networking information that can be used for network monitoring and troubleshooting.

TCP provides reliable connection-oriented services to applications that use it. The term "connection-oriented" refers to a process where two applications using TCP must agree to communicate before any data is exchanged. Timers are kept by both applications and acknowledgements must be received before the timers expire so that information can be retransmitted if lost in transit. Excessive network loading can cause information to be discarded in transit. Checksums provide a measure of data integrity, and sequence numbers allow sequencial delivery of information to upper layer applications.

UDP provides connection-less service. A connection-less service provides no guarantee that information sent ever reaches an intended destination, nor an acknowledgement that such data was received. Examples of applications that use UDP are Domain Name Server (DNS) and Simple Network Management Protocol (SNMP). TCP, UDP, DNS, and SNMP are later discussed in greater detail.

Other protocols such as the Address Resolution Protocol (ARP) or the Reverse ARP (RARP) are used to query the network for MAC address or IP address information. Hosts and routers use the Internet Group Management Protocol (IGMP) for multicasting. IGMP provides a method to let network nodes know who to process multicast information. Sensitive user data is sent using TCP/IP or UDP/IP. The left-hand side of Figure 3 shows the protocol stack associated with data transport in a host machine on NetX. User data that is being sent from a host machine moves from the top to the bottom of the protocol stack. Data moves from bottom to the top of the stack when received by a host machine. To the right of each protocol are encapsulation details, which describe what each protocol sends or expects to receive.

TCP or UDP information is encapsulated into an IP "packet," which is in-turn encapsulated into an 802.3 "frame" before it is sent. The reverse of the preceding procedure is true for a packet that is received.

Table 2 is a review of each field that is used to enable the correct delivery of information to/from an Ethernet application. Note that UDP, and TCP layers both have count values start at number 16; this is done to compare (between TCP and UDP) the total number of fields in Figure 3 enabling data delivery to the proper destination.

A summation of fields involved with correct data delivery

802.3 MAC Layer:

1. Check to see if the 48-bit destination hardware address (MAC address) is it's own, a broadcast address, or a multicast address that it was specifically configured to receive.
2. Check the length field and compare it to how many bytes (MAC layer client data field) there are up to but not including the Cyclic Redundancy Check (CRC) field.
3. Check that the 32-bit CRC matches what is calculated for the frame.

802.2 LLC and SNAP Layers:

4. Check that the DSAP, SSAP, Control, and Org Code default values are correct.
5. Check that the Type field corresponds to a protocol that it supports. (ex., IP, ARP, or RARP)

IP Layer:

6. IP version information corresponds to IP Version 4.
7. Header length (number of 32 bit words in header, including any options).
8. Total length of IP Datagram.
9. 32 bit Datagram number

Table 1. Protocol Stack Check Summary

10. Flag bits
11. Fragment Offset
12. Time to Live
13. Source Protocol ID
14. IP header checksum
15. Destination IP address

TCP Layer:

16. 16 bit source port number
17. 16 bit destination port number
18. 32 bit sequence number
19. 32 bit acknowledgement number
20. TCP header length
21. Special flag bits
22. 16 bit checksum that is calculated over the IP header starting at the source IP address, the TCP header, and data
23. Urgent Pointer

UDP Layer:

16. 16 bit source port number
17. 16 bit destination port number
18. 16 bit UDP Datagram length
19. 16 bit checksum that is calculated over the IP header starting at the source IP address, the UDP header, and data

Table 2. Protocol Stack Check Summary (Continued)

B. NETWORK SERVICES

Data delivery across an Ethernet network is accompanied by addressing services such as the Domain Name Server (DNS), Address Resolution Protocol (ARP), and Proxy ARP. Routing services provide a capability to deliver Ethernet frames between LAN networks, and/or subnetworks.

1. DNS

Because people typically can remember alphanumeric addresses (e.g., "rugrat1") more readily than numeric names (e.g., "125.45.54.44"), DNS provides a means, and a translation between alphanumeric, and numeric addresses. An application requiring hostname or IP address information generates a request to the "gethostbyname" or "gethostbyaddress" library functions asking for the desired IP address given a host name, or the host name given an IP address. These host library functions generate UDP/IP messages destined for a Domain

Name Server (DNS). The DNS responds with the requested information if it is available. No special data integrity checks are associated with DNS.

2. ARP

Before TCP/IP can set up a connection for the client, or UDP/IP send a packet to a destination, the Link (802.3) layer must have an Ethernet MAC address for the corresponding destination. Each host keeps a table called an ARP table where it keeps IP-MAC address mapping information. If the MAC-IP mapping information is not in its ARP table then a process that is transparent to a user application takes place. The Link layer generates an ARP broadcast frame to every host on its associated local network. A broadcast message contains all 1s in the destination MAC address. The ARP messages ask for the MAC address of the host whose IP address is included in the ARP message. Any host that has the IP address associated with the ARP request, or knows that it needs to forward that packet (i.e., to a slip connected host, with the same local network address), responds directly to the requesting host via an directed (rather than broadcast) ARP frame. Note that an ARP message is appended to an Ethernet 802.3 header, and prepended to the 802.3 32-bit CRC; the ARP message itself has no special data integrity checks over-and-above this.

3. Proxy ARP

When a router on a local network is acting as a Proxy ARP server for a destination associated with a received broadcast ARP message (by looking in its routing table), it responds to the source host. This response contains instructions to send packets destined for the ARP associated IP address with the router MAC address. A unicast packet is then sent by the source host, but includes the router MAC address rather than the real MAC address of the desired destination. The router is configured to receive packets that have its MAC address, and routes them to appropriate destinations. Once a packet is sent, if any router between the source and destination determines that it does not have a routing table default, or IP match to forward the IP packet, it responds with an ICMP "host or network unreachable" message back to the originating host. If all the routers between the source and destination are able to deliver the packets to the final destination, only then can the final destination host receive the message, and respond to the sender.

4. Routing

Every router contains a routing table. This table lists host names, network addresses, associated flags, connections in use, interfaces to use, and a default route entry. When a router receives a request for a packet to be routed it searches this table by name, network address, and then for a default route. If there is not a matching name, matching network address, or default route the router will inform the sender that the destination is not reachable. If a router forwards a packet on a default route then the final router that has the correct destination-host locally attached will forward it to the destination-host, unless of course the host is unreachable.

C. CONNECTION ESTABLISHMENT

Two protocols used to transmit user information between NetX connected Ethernet applications are TCP or UDP.

1. TCP Connection Establishment

TCP is an application connection-oriented protocol. Before a source application can establish a TCP connection it has to ask the destination machine for permission, and if given, must respond with and acknowledgement before any application data is sent. DNS, and/or ARP procedures to acquire addressing information can precede the TCP connection process. Once a source TCP application has an IP-MAC address pair for transmission, the connection establishment procedure begins.

The source sends a TCP/IP packet with only header information to the destination. This header has all fields including destination port, source sequence number (SSN), and a special "synchronize sequence number (SYN)" flag bit. The destination responds with the same type of packet except it increments the SSN by one and returns it to the sender with its own destination sequence number (DSEN), and port number to use. When the source receives this, it responds again to the destination, incrementing the DESN by one. A TCP connection is now setup between the two applications (not the network).

To close the connection, both source and destination applications must send a "finish sending data" TCP/IP packet for the connection to be released by each application. There is no "idle" time-out function for TCP. If two hosts applications open a connection, and if no data is transmitted they stay up; routers, etc., between the devices could go down, but the host applications keep their application level connections. An application may have a timeout function and even use a keepalive function to insure that an inactive connection to a host is operational, else close the TCP connection.

2. UDP Connectionless Transmission

Unlike TCP, UDP is connectionless and provides no guarantee of delivery. Data is simply sent, without any acknowledgement from the intended destination.

III. LAN EMULATION (LANE)

The LANE protocol provides a definition of the architecture, services, functions, frame formats, protocols, and procedures to implement an Emulated LAN (ELAN) over an ATM network. Two versions of LANE exist, LANE version 1 (LANE v1), and LANE version 2 (LANE v2). LANE v1 was standardized in 1995, and defined the interfaces between the LAN Emulation Client (LEC), and the LANE services: (1) LEC Server (LECS), (2) LANE Server (LES), and (3) Broadcast and Unknown Server (BUS). The interoperations, however, between the LECS, LES, and BUS were not specified in LEv1. In 1997, the ATM Forum created the LAN Emulation over ATM - Local User Network Interface version 2 (LUNI v2) specification. The LUNI v2 specification superceded the 1995 LANE v1 specification, but it still did not define the interoperations between the LESs. The development of the Server Cache Synchronous Protocol (SCSP RFC 2334) in 1998, and work on the UNI 4.0 protocol, lead to the release of the LANE – Local Network to Network Interface version 2 (LNNI v2) specification in February of 1999. The LNNI v2 specification provided LANE service (LECS, LES, and BUS) interoperation specifications, a new LANE service called the Selective Multicast Server (SMS) service, as well as a host of other capabilities that were desperately needed given the fast pace of related network systems development, and system interoperation needs. LANE v2 refers to both LUNI v2, and LNNI v2 specifications.

A. LANE CLIENT AND SERVICES

Each emulated LAN is composed of a set of LECs, and a single LANE service consisting of LECS, LESs, BUSs, and possibly Selective Multicast Servers (SMS). A LEC can reside on any NetX ATM device such as an AED, AS, or AES in Figure 1. A LEC represents a single user, or a set of users, identified by ATM, or MAC address(es). LANE considers ATM addresses with different "Selector" bytes as different ATM addresses. The LEC performs data forwarding, address resolution, and MAC level 802.3 emulation for upper layer applications using the LUNI protocol to interface with the LANE services within an ELAN.

The LECS, which also can reside on any AED, AS or AED in NetX, provides the assignment of LECs to individual emulated LANs. It does this by providing a LEC with an ATM address of a LES corresponding to the ELAN the client is authorized to join. The LES provides a method of registering, and resolving ATM addresses associated with Ethernet MAC addresses. Each LEC registers LAN Ethernet MAC addresses it represents with the LES. The BUS handles data that needs to be broadcast over an ELAN. Unicast, multicast, and broadcast frames are sent to a BUS. Typically unicast frames are not sent to a BUS once an ATM destination address is found for the true destination and a data circuit is established. A SMS corresponds to a LANE service; the SMS specification was developed too late to be incorporated into the LUNI specification, so it was included within the LNNI specification. An SMS is discussed within Section III.E LNNI v2 of this document.

B. LANE VCCS

Virtual Circuit Connections (VCCs) are ATM connection-oriented communication paths setup between clients and servers, and between servers. Figure 4 shows an overview of VCCs associated with LANE. VCC types are broken into LUNI VCCs, or LNNI VCCs. LUNI VCCs connects clients to LANE services, and LNNI VCCs interconnect LANE services.

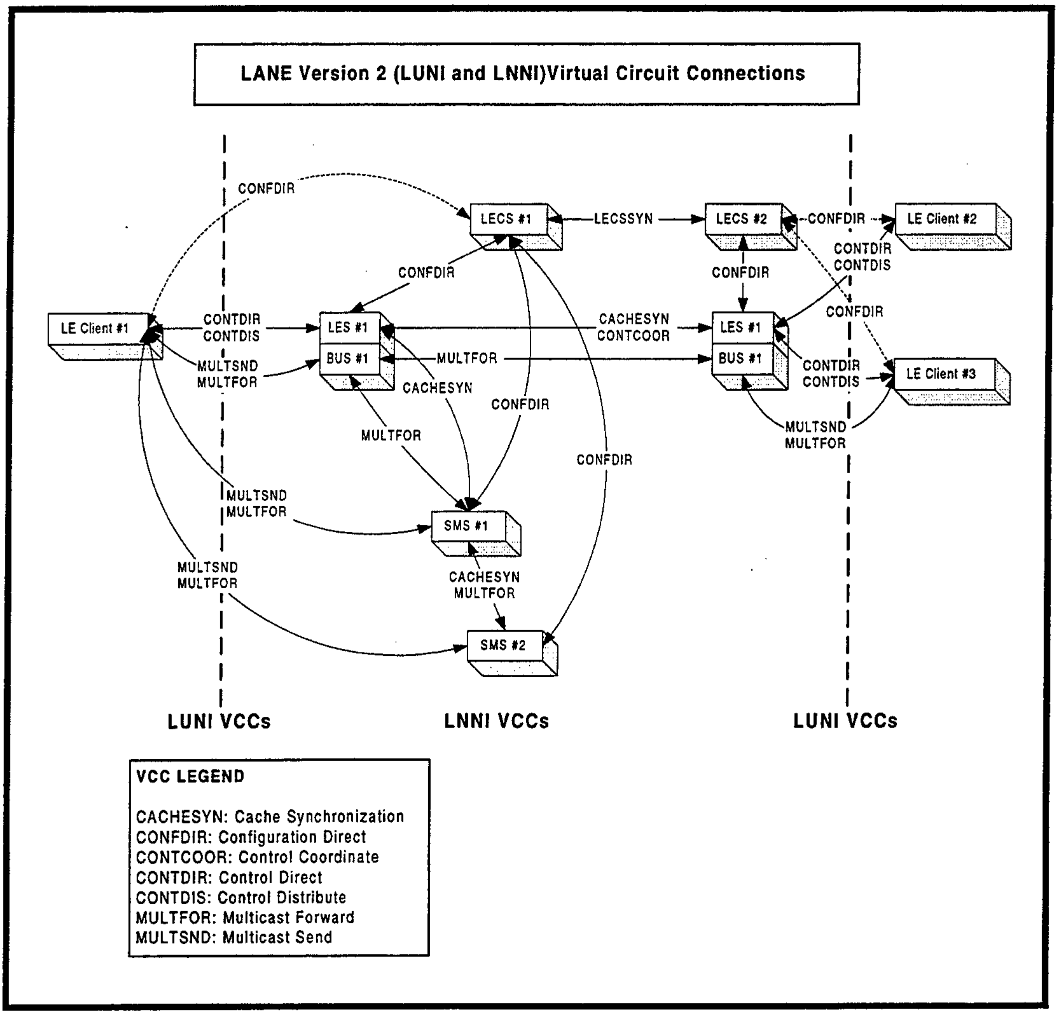


Figure 4. An Overview of VCCs associated with LANE

C. LANE COMPONENT IDENTIFICATION

When either the LANE Layer Management (a management entity within a LANE device), higher layer protocols above a LEC layer, or the LANE Connection Management (a connection management entity within a LANE device) requires a VCC to be setup, software primitives are exchanged with the UNI software, and a UNI signaling message is encoded and sent. UNI signaling messages contain Broadband Low-Layer Information

(BLLI) Information Elements (IE) which describe the type of VCC requested (LLC-multiplexed VCC, Control or Configuration VCC, or Data Direct VCC), as well as information pertaining to layers 2 and 3 of the protocol stack (See Figure 19 and Figure 20). The BLLI information in the setup message is bound to the VCC after it is setup. LANE uses a combination of ATM address (including the Selector byte), BLLI setup information, and LLC code points to distinguish between VCC types, and LANE entities. LLC-multiplexed VCCs allow multiple protocols (of which LANE is one) to operate over it by prepending a 3-byte ANSI/IEEE 802.2 header to each ATM Adaptation Layer number 5 (AAL5) data frame.

D. LUNI V2

LUNI v2 is the first half of the LANE v2 protocol. A description of the VCCs, frame formats, LUNI connectivity, registration, and termination procedures, as well as an overview of the LANE address resolution process follows. Some ATM terms found in this section, such as LLC-multiplexing, are described in Chapter IV.

1. LUNI v2 VCC Descriptions

The following three sections, Control VCCs, and Data VCCs, and LAN Emulation Control Frame are provided as an aid for further reading.

a. LUNI v2 Control VCCs:

LUNI v2 control VCCs (See Figure 4) are connections that carry only control messages. They do not carry end user data traffic, and are not LLC-multiplexed. An overview of the LUNI v2 control VCCs is as follows:

1. Configuration Direct (CONFDIR)
    - Point-to-point VCC only.
    - Bi-directional.
    - Setup in LEC to LECS in connect phase.
    - Can be released while participating in ELAN.
2. Control Direct (CONTDIR)
    - Point-to-point VCC only.
    - Bi-directional.
    - Setup by LEC to LES in initialization phase.
    - Cannot be released while participating in ELAN.
3. Control Distribute (CONTDIS)
    - Point-to-multipoint VCC only.
    - Uni-directional.
    - LEC is required to allow this to be optionally setup by LES in initialization phase.
    - Cannot be released (if setup) while participating in ELAN.

b. LUNI v2 Data VCCs:

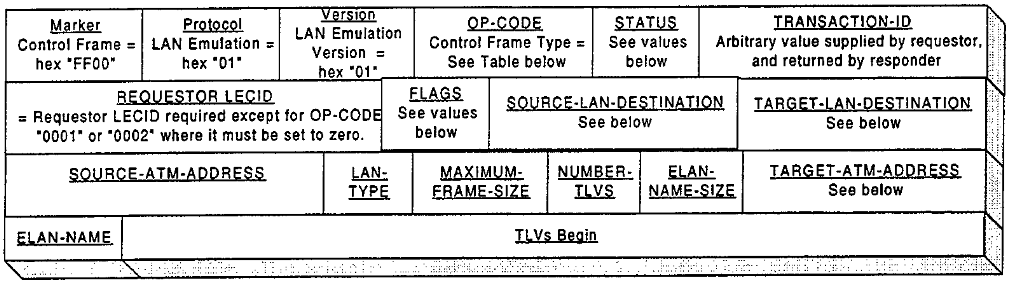
LUNI v2 Data VCCs (See Figure 4) are connections that carry data, rather than control type traffic. Data VCCs can be non-multiplexed, or LLC-multiplexed (dependent on the VCC type). Data VCCs are as follows:

4. Default Multicast Send (DEFMULSND)
    - Point-to-point VCC only.
    - Bi-directional.
    - Setup by a LEC to a BUS. It is used for initially sending unicast traffic (when the unicast ATM address of the real destination's LEC is not known), broadcast traffic, and multicast traffic (when an alternate multicast address has not been resolved).
    - A specified QOS can be requested.
    - Can be non-multiplexed or LLC-multiplexed.
5. Selective Multicast Send (SELMULSND)
    - Point-to-point VCC only.
    - Bi-directional.
    - Setup by a LEC to a BUS, and/or a SMS.
    - A specified QOS can be requested.
    - Must be non-multiplexed.
    - Multicast Forward (MULFOR).
    - Point-to-multipoint VCC only.
    - Uni-directional.
    - Setup by a BUS and/or a SMS to a LEC.
    - At least one MULFOR must be setup by a BUS, and maintained, for the duration of a LEC ELAN participation.
6. Data Direct (DATDIR)
    - Point-to-point VCC only.
    - Bi-directional.
    - Setup by LEC to other LECs, or to the BUS and/or SMS (in the form of multicast send VCCs).
    - A specified QOS can be requested.
    - Can be non-multiplexed, or LLC-multiplexed.

2. LANE Control Frame Format

The previous section provided an overview of both control VCCs, and data VCCs used by LANE. Control VCCs transport LANE control frames which are formatted according to Figure 5. A specific message type carried by a control frame is identified by an OP-CODE field value setting within a control frame. LANE uses UNI signaling to setup contol VCCs to transport control frames. LANE control frames are transported like LE data frames (See Figure 6) in that they interface directly with the CPCS layer (See Figure 16).

LANE Control Frame (LLC multiplexed Flush Control frame (exception)is not shown)



Marker
Control Frame = hex "FF00"

Protocol
LAN Emulation = hex "01"

Version
LAN Emulation Version = hex "01"

OP-CODE
Control Frame Type = See Table below

STATUS
See values below

TRANSACTION-ID
Arbitrary value supplied by requestor, and returned by responder

REQUESTOR LECID
= Requestor LECID required except for OP-CODE "0001" or "0002" where it must be set to zero.

FLAGS
See values below

SOURCE-LAN-DESTINATION
See below

TARGET-LAN-DESTINATION
See below

SOURCE-ATM-ADDRESS

LAN-TYPE

MAXIMUM-FRAME-SIZE

NUMBER-TLVS

ELAN-NAME-SIZE

TARGET-ATM-ADDRESS
See below

ELAN-NAME

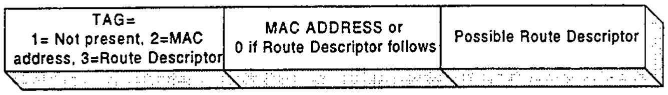
TLVs Begin

OP-CODE Values		OP-CODE Values	
Value	Function	Value	Function
hex"0001"	LE_CONFIGURE_REQUEST	hex"0107"	LE_FLUSH_RESPONSE
hex"0101"	LE_CONFIGURE_RESPONSE	hex"0008"	LE_NARP_REQUEST
hex"0002"	LE_JOIN_REQUEST	hex"0108"	Undefined
hex"0102"	LE_JOIN_RESPONSE	hex"0009"	LE_TOPOLOGY_REQUEST
hex"0003"	READY_QUERY	hex"0109"	Undefined
hex"0103"	READY_IND	hex"000A"	LE_VERIFY_REQUEST
hex"0004"	LE_REGISTER_REQUEST	hex"010A"	LE_VERIFY_RESPONSE
hex"0104"	LE_REGISTER_RESPONSE	hex"000b"	LNNI_CONFIGURE_TRIGGER
hex"0005"	LE_UNREGISTER_REQUEST	hex"000c"	LNNI_LECS_SYNC_REQUEST
hex"0105"	LE_UNREGISTER_RESPONSE	hex"000d"	LNNI_KEP_ALIVE_REQUEST
hex"0006"	LE_ARP_REQUEST	hex"010d"	LNNI_KEEP_ALIVE_RESPONSE
hex"0106"	LE_ARP_RESPONSE	hex"000e"	LNNI_VALIDATE_REQUEST
hex"0007"	LE_FLUSH_REQUEST	hex"010e"	LNNI_VALIDATE_RESPONSE

STATUS Values	
Code(dec)	Meaning
0	Successful response
1	VERSION field of request contains a higher value than supported by the responder
2	The parameters given are incompatible with the ELAN
4	SOURCE-LAN-DESTINATION duplicates one previously registered
5	SOURCE-ATM-ADDRESS duplicates a previously registered ATM address
6	Responder is unable to grant request (insufficient space or ability to establish VCCs)
7	Request denied for security reasons
8	LECID field is not zero, or is not the LECID of client
9	LAN Destination is a multicast address, or Route Descriptor on an 802.3 LAN
10	Source or Target ATM address not in a recognizable format or not valid
20	LEC is not recognized
21	Parameters suplied give conficting answers, or refuse service without a reason
22	LEC has not provided sufficient information
24	TLV not found

FLAGS Values		
Code(hex)	Name	Use
"0001"	Remote Address	LE_ARP request or response
"0002"	V2 Capable	LE_CONFIG_REQUEST, LE_JOIN_REQUEST
"0004"	Selective Multicast	LE_JOIN_REQUEST
"0008"	V2 Required	LE_JOIN_RESPONSE
"0080"	Proxy Flag	LE_JOIN_REQUEST
"0100"	Topology Change	LE_TOPOLOGY_REQUEST
"0200"	Token Ring Eplorer Exclude	LE_JOIN_REQUEST
"0400"	IS_LE_SERVER	LNNI - LE_CONFIG_REQUEST

Lan Destination Fields



TAG=
1= Not present, 2=MAC address, 3=Route Descriptor

MAC ADDRESS or
0 if Route Descriptor follows

Possible Route Descriptor

Figure 5. LANE Control Frame Format

3. LANE Data Frame Format

LUNI v1 only supports non-multiplexed LANE data frame formats. LUNI v2 and LNNI v2 both support the non-multiplexed, and LLC-multiplexed data frame formats. The format for a non-multiplexed and LLC-multiplexed data frame is shown in Figure 6.

An LLC-multiplexed VCC can be used by many different protocols, not just LANE. Each LLC-multiplexed data frame is prepended with a 3-byte ANSI/IEEE 802.2 LLC header to distinguish protocol (flow) types and ELANs associated with each flow. A flow is data or traffic for a single ELAN or protocol. When a VCC is being setup, a UNI SETUP signaling message is encoded with a BLLI IE. The BLLI IE distinguishes the requested VCC type as either an LLC-multiplexed VCC, or non-multiplexed VCC. The LLC-multiplexed VCC is identified in the "User Information Layer 2 Protocol Information" field (See Figure 20). The ELAN-ID is specific to LANE v2, and is required. The "User Information Layer 3 Protocol Information" identifies the type of non-multiplexed VCC requested (See Figure 20).

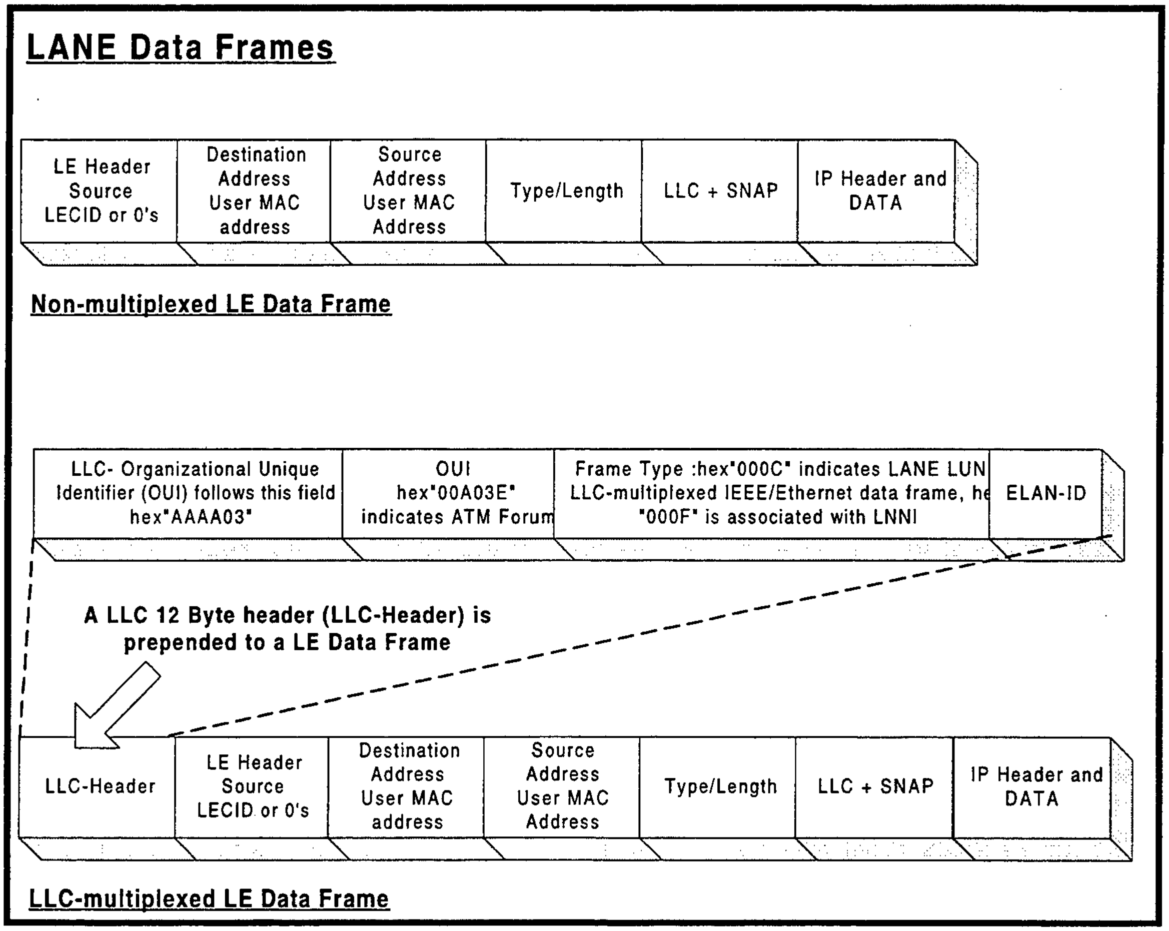

LANE Data Frames
LE Header Source LECID or 0's
Destination Address User MAC address
Source Address User MAC Address
Type/Length
LLC + SNAP
IP Header and DATA
Non-multiplexed LE Data Frame
LLC- Organizational Unique Identifier (OUI) follows this field hex"AAAA03"
OUI hex"00A03E" indicates ATM Forum
Frame Type :hex"000C" indicates LANE LUN LLC-multiplexed IEEE/Ethernet data frame, h "000F" is associated with LNNI
ELAN-ID
A LLC 12 Byte header (LLC-Header) is prepended to a LE Data Frame
LLC-Header
LE Header Source LECID or 0's
Destination Address User MAC address
Source Address User MAC Address
Type/Length
LLC + SNAP
IP Header and DATA
LLC-multiplexed LE Data Frame

Figure 6. LANE Data Frame Format

4. LUNI Phases of Connectivity

The LUNI phase of connectivity starts with an LEC and LES in an initial state, each having various parameters that are either already pre-configured or need to be configured in order to complete the connectivity process. There are five phases associated with connectivity before a LEC can reach an operational state: (1) Connect, (2) Configuration, (3) Join, (4) Registration, and the (5) Bus Connect phase. The Registration Phase section in this thesis also covers De-registration, because this phase occurs whenever a new EES device connects or disconnects to/from a LEC (acting on its behalf) after the LEC competes the Join Phase of LUNI connectivity.

The first two phases are the LECS Connect phase, and the Configure phase; these phases are entered if a LEC does not have a pre-configured valid operational LES address. The Connect phase involves acquiring a valid operational LECS address, and setting up a CONFDIR VCC to it. See Figure 4 for VCC placement. See Figure 5 for all control frame (except LE_FLUSH) structures.

a. Connect Phase

This first phase starts by acquiring a valid operational LECS address. Three LECS ATM address resolution methods are given in order of precedence: (1) The LEC may have a valid operational LECS ATM address pre-configured, (2) it may use a UNI ILMI process to get it, or (3) use an ATM Forum "well-known" LECS address (this method can fail if the UNI implementation doesn't support "well-known" address registration, or if IISP is used and routes the "well-known" address to the wrong destination) [4]. If these address enquiry attempts fail then a LEC will wait a pre-configured retry time before starting this phase again. If it does get a valid operational LECS address then it sets up a CONFDIR VCC to it, and moves to the second connectivity phase (Configuration phase).

b. Configuration Phase

The Configuration phase involves querying the LECS, via "LE_CONFIGURE_REQUEST" control frame for an assigned operational LES ATM address, and possibly other parameters associated with an ELAN it wishes to join. The CONFDIR VCC is bi-directional and is used for LECS replies via "LE_CONFIGURE_RESPONSE" control frames. See Figure 4 for VCC placement. See Figure 5 for all control frame (except LE_FLUSH) structures.

If a LECS is pre-configured with LAN type, maximum data frame size, ELAN name, LES ATM address, V2 Capability flag setting, and selective multicast registration information, then it can skip the Configuration phase and move to the Join phase. If it does not know this information then it sends the LE_CONFIGURE_REQUEST with a specified TRANSACTION-ID over the CONFDIR VCC to the LECS for it.

The legality and validity of the parameters in the LE_CONFIGURE_REQUEST are checked by the LECS. If by checking the configuration file in the LECS, or checking via some other means (not specified), results in a determination that the requesting LEC is not authorized to join an ELAN, or if the parameters in the request were not configured correctly, it will return an LE_CONFIGURE_RESPONSE with a STATUS field set to "1" (No Success). If a LEC were configuring itself, as opposed to a proxy LEC configuring on the behalf of other LECs, it would then go back to the beginning of the LUNI connectivity procedures and start again after a pre-determined reconfiguration delay.

The calling party's ATM address in the LE_CONFIGURE_REQUEST can not be used to determine which ELAN the LEC can join, it can only be used to discern whether or not to reply and/or release the connection.

If a LE_CONFIGURE_REQUEST is not responded to within an initial Control time-out period, it is retried after a (Control time-out period multiplied by the control time-out multiplier) time. After not receiving a response within the maximum Control time-out period then the LEC requesting connectivity must go back to the beginning of the connectivity process.

If the LECS determines that the LEC can join an ELAN, it will return a LE_CONFIGURE_RESPONSE with the same TRANSACTION-ID of the request, a STATUS field set to 0 (Success), supply the above parameters, as well as the LES ATM address for the ELAN the client had been given permission to join. This response (if the LES is a LANE v2 server) will include a TLV with an ELAN-ID value that must be included in all LLC-multiplexed data frames sent or received on the ELAN. Note that the ELAN-ID value could also be returned in the Join phase of connectivity if the Configure phase were skipped due to pre-configuration of the LEC. Several other TLVs (if present) in the response dictate the LEC possibly setting the maximum data frame size value. Note that vendor-specific TLV extensions may be implemented in LE_CONFIGURE_REQUESTs, and LE_CONFIGURE_RESPONSEs using the OUI of the vendor. Undocumented extensions may provide a back door capability that is known only by the vendor. The LEC releases (optional) the CONFDIR VCC after this phase is completed and move to the Join phase of the connectivity process.

c. *Join Phase*

The Join phase involves the LEC setting up a point-to-point, bi-directional, Control Direct VCC to the LES (using normal UNI signaling procedures with B-LLI information), transmitting a LE_JOIN_REQUEST control frame to the LES (over the newly formed Control Direct VCC), accepting a point-to-point Control Distribute uni-directional VCC from the LES (using the same UNI procedure as the Control Direct VCC setup), and receiving a LE_JOIN_RESPONSE from the LES. See Figure 4 for VCC placement. See Figure 5 for all control frame (except LE_FLUSH) structures.

The LE_JOIN_REQUEST must contain the variables: (1) LAN type, (2) Proxy, (3) ELAN name, (4) V2 Capable flag, (5) Selective Multicast Configured flag, (6) Token Explorer Frame Exclude Flag, (7) LEC Primary ATM Address, (8) Maximum Frame Size, and optionally (9): Layer-3-Address, X5-Adjustment, or Preferred-LES TLVs if LANE v2 is supported.

The LES checks the LE_JOIN_REQUEST to make sure that: (1) the Maximum Frame Size, and V2 Capable flags have valid settings, (2) that the unicast MAC in the request does not pertain to a multicast address, or a previously registered MAC address, and (3) the REQUESTER-LECID is zero. The LAN type and Maximum Data Frame Size can be "Unspecified."

A LES has an option to validate a LEC with a LECS before allowing ELAN membership. This process involves the LES relaying the LE_JOIN request to the LECS with changed OP-CODE and TARGET-ATM-ADDRESS field. The LECS checks through an unspecified means and responds with a success or failure indication. The LES then notifies the LEC of the failure.

If the above conditions result in a failed join attempt a LES responds with a LEC_JOIN_RESPONSE frame, whose STATUS field indicates a failed join attempt.

Once a LE_JOIN_REQUEST is sent, a LES must establish a unicast point-to-point Control Distribute VCC connection to the primary address of the LEC before sending a LE_JOIN_RESPONSE. The LEC must accept this VCC. The LES returns a LE_JOIN_RESPONSE to the LEC. If any response from a LEC indicates a failure then the LES must terminate its ELAN membership. If the LES responds to the LEC with a join response, indicating a successful join, the LEC must update the following variables, or terminate its ELAN membership: (1) LAN type, (2) ELAN name, (3) V2 Capable flag, (4) Selective Multicast Configured flag, (5) Maximum Data Frame Size, (6) LECID, (7) Local Segment, and (8) ELANID.

Like the LEC configure requests and responses, the LEC join requests and responses follow the same procedures for retrying to establish membership and ELAN termination as well as those procedures associated with the Control time-out, Control time-out multiplier, and associated Maximum Control time-out periods. If either of the control VCCs between the LEC and LES fail then either or both the LEC and LES will initiate ELAN termination; the LEC would then return to the beginning of the LUNI connectivity procedure. Termination of ELAN membership includes the termination of all Control VCCs between the LEC and LES.

d. Registration Phase

This phase involves the registration, or de-registration of local MAC addresses, ATM addresses, and any associated address Type/Length/Value (TLV) bindings with the LES after the Join phase.

TLVs are information units that are passed within LE control frames. An example of a TLV, required by a LE_CONFIGURE_REQUEST control frame, is "Layer-3-Address" for LANE v2. An example of a TLV, required by a LE_CONFIGURE_RESPONSE control frame, is "ELAN-ID." Lists of required TLVs for each control frame are found in the LUNI specification.

The Registration phase utilizes the Control Direct, and Control Distribute VCCs between the LEC and the LES. See Figure 4 for Control Direct, and Control Distribute VCC placement. See Figure 5 to view the format for all LANE control frame (except LE_FLUSH) structures.

A LEC issues a LE_REGISTER_REQUEST to a LES to register a MAC address – ATM address binding along with any associated TLVs. Multiple MAC addresses may be bound to a single ATM address, however only one LLC-multiplexed, and/or non-multiplexed ATM address may be associated with a single MAC address registered. For each LE_REGISTER_REQUEST sent, an LE_REGISTER_RESPONSE is returned indicating a successful or failed address registration.

A non-response to a LE_REGISTER_REQUEST result in retries, with intervals between retries, and maximum Control time-out periods utilized. If a LE_REGISTER_REQUEST is not responded to within the maximum Control time-out then the LEC must terminate its ELAN membership, and return to the beginning of the LUNI connectivity procedure.

A LEC sends a LE_UNREGISTER_REQUEST to a LES to de-register a MAC address – ATM address binding along with any associated TLVs. A LES matches REQUESTOR_LECID, and LAN_DESTINATION information of all received LE_UNREGISTER_REQUEST frames with LEC address registration bindings before it deletes all associated addresses and TLV bindings. This prevents a LEC from de-registering another LEC from the LES registration database.

Any LE_REGISTER_RESPONSE to a non-proxy LEC indicating a failed attempt to register must result in the LEC terminating its ELAN membership. Proxy LEC local registrations can fail and keep their ELAN membership provided that the cause of the failure in the response states the failure was due to the LES having "insufficient resources to grant the request."

LANE address registration control is provided by a LES. A diagram showing LANE address registration control is shown in . This diagram represents a time sequence of connection requests starting with LEC 1, and ending with LEC 7.

A LES will not allow: (1) different LECs to register the same MAC address, (2) different LECs to register the same non-multiplexed ATM address, and (3) different LECs to register the same LLC-multiplexed ATM address. A LES will allow: (1) different LECs to register the same ATM address only if they are of different multiplexing schemes, and (2) a LEC to register one LLC-multiplexed ATM address, and/or one non-multiplexed ATM address with only one MAC address

There are no provisions to check duplicate ATM, or MAC address registrations between ELANs. Failed registrations due to a pre-registered ATM addresses are sent a LE_REGISTER_RESPONSE with a cause in its status field.

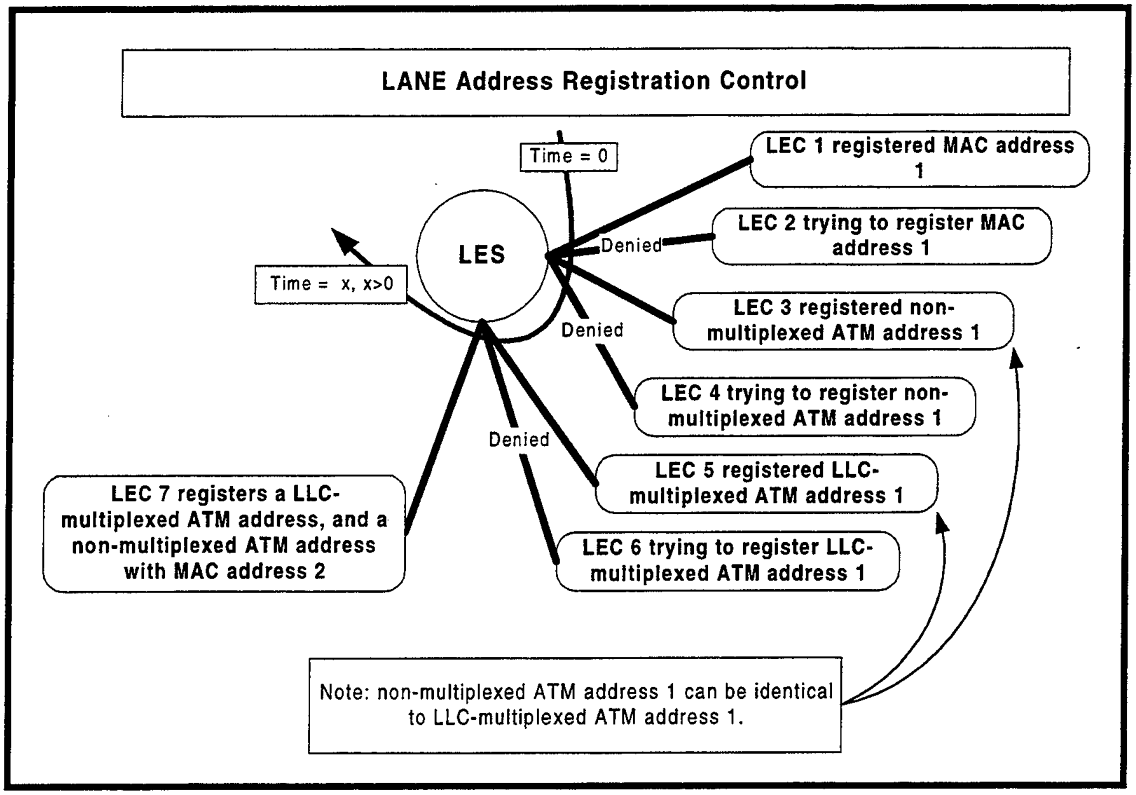

LANE Address Registration Control
Time = 0
LEC 1 registered MAC address 1
LEC 2 trying to register MAC address 1
Denied
LES
Time = x, x>0
LEC 3 registered non-multiplexed ATM address 1
Denied
LEC 4 trying to register non-multiplexed ATM address 1
Denied
LEC 5 registered LLC-multiplexed ATM address 1
LEC 7 registers a LLC-multiplexed ATM address, and a non-multiplexed ATM address with MAC address 2
LEC 6 trying to register LLC-multiplexed ATM address 1
Note: non-multiplexed ATM address 1 can be identical to LLC-multiplexed ATM address 1.

Figure 7. LANE Address Registration Control

e. Bus Connect Phase

This phase includes the LEC sending an LE_ARP_REQUEST to the LES with the MAC broadcast of "all-ones," receiving an LE_ARP_RESPONSE with the ATM address of the BUS, then setting up a bi-directional Multicast Send VCC to the BUS. The BUS responds to the Multicast Send frame by setting up one, or more point-to-multipoint Multicast Forward VCCs. These Multicast Forward VCCs may not come from the BUS addressed in the LE_ARP_REQUEST. The LUNI v2 specification provides a LE_VERIFY protocol for a LEC to check on Multicast Forward VCC source ATM addresses, before allowing Multicast Forward VCCs to be setup. Note that the use of the LE_VERIFY protocol is optional. A client could follow a BUS connect process when trying to establish Selective Multicast Send (established by the LEC), and Selective Multicast Forward (established by the SMS) VCCs. Selective Multicast Forward (Send, or Receive) VCCs is discussed in the LNNI v2 part of this paper. If a host does not register selective multicast addresses then it will receive all broadcast and multicast frames from the BUS. Registering selective multicast addresses allows a host to limit the multicast frames that it sends. A SMS, discussed in Section III.E LNNI v2, off-loads LANE v2 multicast traffic processing from the BUS.

5. LANE Connection Management Services for LECs and LESs

Connection management services from a layer perspective sit below a LEC layer and any upper layer LLC or Bridge layers. It is responsible for the reception or transmission of call setups, releases, and add or drop party signals. It manages flows (a group of connections associated with a particular protocol, on a VCC), not each connection associated with non-lane protocol flows on LLC-multiplexed data direct VCCs. The LLC-multiplexed VCCs do their own management. A non-multiplexed data direct VCC timeout variable (C12 -set to 20 minutes by default) is an example of a management variable used for connection management. When a non-multiplexed data direct VCC is idle for a time that meets the C12 time, the VCC is released. Note that the maximum time for this variable can be set to "Unlimited." A listing of reasons why LANE connection management would terminate VCCs on a LEC, or LES are given in the next section. Following the VCC Termination section, is an overview of connection management messages associated with address discovery.

6. VCC Termination

The following is a reference for VCC termination causes.

a. Causes for Multicast Forward VCC termination

1. A failed Multicast Send VCC. (optional).
2. A failed Multicast Send VCC that causes the LEC to restart the BUS Connect Phase, which results in a different BUS ATM address.

b. Causes for Multicast Send VCC termination

1. A failed (last one) Multicast Forward VCC for a period longer than the LEC "Forward Disconnect Time-out" variable time.

c. Causes for LEC ELAN termination

1. Resolution of BUS ATM address fails.
2. A failed attempt to reconnect to a BUS.
3. The release of a Control Direct and/or a Control Distribute VCC.
4. If the BUS does not see that a Multicast Send VCC was reestablished within its "Send Disconnect Time-out" variable time. (optional)
5. If the BUS fails to setup a Multicast Forward VCC to a LEC that has a Multicast Send VCC connection to the LEC. (optional)

d. VCC Losses that do not Result in ELAN Termination

1. Loss of a Selective Multicast Send VCC results in no action if the LEC does not have anything more to send.
2. Loss of a Multicast Forward VCC results in no action if at least one other Multicast Forward VCC exists to the BUS.

7. Address Resolution Protocol, Procedures and Frame Formats

A LEC sends a LE_ARP_REQUEST message to its LES to discover addressing information of other ELAN members. A returned message, either providing the requested information or failure indication is made via a LE_ARP_RESPONSE message.

Any LEC that receives a LE_ARP_REQUEST with a LAN destination that matches any of its local variables: (1) Local Unicast MAC Address(es), (2) Local Route Descriptor(s), (3) Remote Unicast MAC Address(es), or (4) Remote Route Descriptors, must respond. A LES responds to Multicast MAC address, and BUS MAC address requests with associated ATM address, and TLV values as well.

A LES does not forward LE_ARP REQUESTS if it responds to the requesting LEC on behalf of a registered requested address. All TLVs associated with the MAC/ATM address pair are included in a LE_ARP_RESPONSE.

All LE_ARP_REQUESTS associated with unregistered MAC/ATM address pairs are not forwarded to local LECs, nor other LESs, they are only forwarded to Proxy LECS associated with an ELAN. Note that this does not include Targetless LE_ARP_REQUESTS.

When a LEC wishes to change a MAC/ATM address binding, or associated TLV information it may issue a Targetless LE_ARP_REQUEST message. A Targetless LE_ARP_REQUEST message is like a broadcast message that, unlike LE_ARP_REQUESTS is sent to all members of the ELAN. This message does not have a specific LEC TARGET-MAC-ADDRESS, but it does have SOURCE-MAC address and bound SOURCE_ATM_DESTINATION address information. All LECS receiving this message must delete any cache information associated with the SOURCE_LAN_DESTINATION, and may replace this information with what is received. Clients that do not have the SOURCE_LANE_DESTINATION in their caches may ignore it.

Two other frames, the LE_NARP_REQUEST, and the LE_TOPOLOGY_REQUEST frame are used for address binding, and address binding aging changes respectively. The LE_NARP_REQUEST is used to tell other clients that specific LE ARP cache MAC/ATM address binding information is no longer valid, and that they must delete the binding, and possibly replace it with what is in the received LE_NARP_REQUEST message. The LE_NARP_REQUEST message is superceded in LANE v2 with the Targetless LE_ARP_REQUEST. These messages relate to local as well as remote addresses. LANE v1 LE_NARP_REQUEST messages are handled the same as LANE v2 Targetless_LE_ARP messages. Both LE_NARP_REQUEST and Targetless LE_ARP_REQUESTs are forwarded to all LECs in the ELAN by the LES.

A LE_TOPOLOGY_REQUEST is used with IEEE 802.1D transparent bridging (Proxy LEC) for topology change information promulgation. A LE_TOPOLOGY_REQUEST message is one that a Proxy LEC sends to a LES (which then forwards it to all LECs) whenever a bridge needs to broadcast to the BUS Configuration Bridge Protocol Data Units (BPDUs) to age out bridge related information quickly. The receiver of a BPDU message (from a BUS) will also get a LE_TOPOLOGY_REQUEST message from a LES. According to the LUNI v2 specification a BPDU may take precedence over a LE_TOPOLOGY_REQUEST when address binding information when bridge table address information is combined with the LE ARP table information.

8. Data Transfer Protocol and Procedures

Unicast data frames must be sent on Data Direct or Default Multicast Send VCCs. Multicast Data frames must be sent on Default or Selective Multicast Send VCCs. A LEC must insert its own LECID or hex "0000" in the LE_HEADER of every data frame, but it is required to ignore LEC-IDs on received data frames, and the option to filter on frames received on Multicast Send or Multicast Forward VCCs for its own LECID. All LLC-Multiplexed VCCs received with an incorrect ELAN-ID (LANE v2 only) are ignored; therefore, due to LLC and ELAN-ID checking, registered LEC MAC addresses will never receive a frame destined for another ELAN.

A LEC is not required to discard frames with incorrect B-LLI values. This is to enable frames to carry data destined for protocols other than LANE. A VCC Time-out period is only (as an option) to be used for non-multiplexed VCC data-direct connections; It is not to be used for LLC-Multiplexed VCCs.

9. Connection Setup

LUNI v1, and v2 use UNI signaling procedures to establish VCCs; however, two additional control messages are added (READY_IND and READY_QUERY) to the connection setup process. Figure 8 shows the sequence of messages that are sent between two LECs that are trying to setup a data direct connection. This data direct connection traverses two ATM switches. Each message is numbered to show the order in which they are sent, and the protocol (UNI, PNNI, or LANE) used to create or process the messages. Above each vertical line in the figure are references to protocol procedure figures in this document, which specify UNI and PNNI procedures at greater detail. Normally in UNI, an originating LEC (LEC A in Figure 8) sends a SETUP message that eventually reaches the designation (LEC B in Figure 8). When LEC B accepts the connection it sends a CONNECT message back to LEC A, but instead of the LEC A sending an acknowledgement CONNECT-ACK back to LEC B (to indicate that it is ready to receive data), the ATM switch (Switch 2 in Figure 8) does. The end result is that LEC B (using only UNI procedures) may send data to LEC A even though LEC A may not have received the CONNECT message, and therefore is not able to accept data on the new VCC. To fix this problem LUNI v2 requires that LEC B wait for a READY_IND message from LEC A on the newly established VCC before it starts to send information on it. A timer (C28) is specified with a READY_QUERY message defined in the event of a non-response. If a READY_IND or (optionally) data is not received on a newly established VCC before the (C28) expires a READY_QUERY message is sent to LEC A. LEC A must respond to READY_QUERY messages with a READY_IND message.

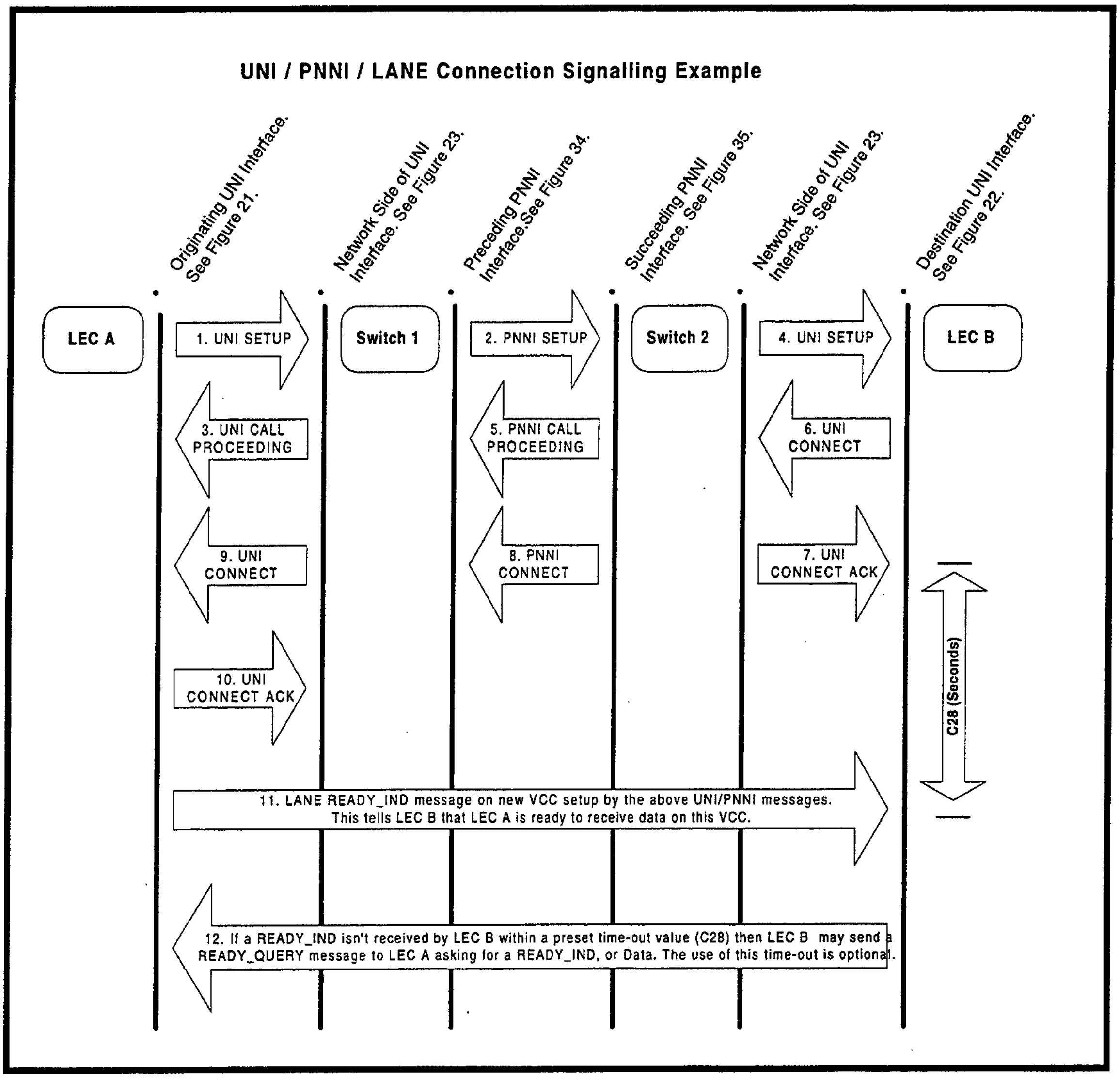

UNI / PNNI / LANE Connection Signalling Example
Originating UNI Interface. See Figure 21.
Network Side of UNI Interface. See Figure 23.
Preceding PNNI Interface.See Figure 34.
Succeeding PNNI Interface. See Figure 35.
Network Side of UNI Interface. See Figure 23.
Destination UNI Interface. See Figure 22.
LEC A
Switch 1
Switch 2
LEC B
1. UNI SETUP
2. PNNI SETUP
4. UNI SETUP
3. UNI CALL PROCEEDING
5. PNNI CALL PROCEEDING
6. UNI CONNECT
9. UNI CONNECT
8. PNNI CONNECT
7. UNI CONNECT ACK
10. UNI CONNECT ACK
C28 (Seconds)
11. LANE READY_IND message on new VCC setup by the above UNI/PNNI messages. This tells LEC B that LEC A is ready to receive data on this VCC.
12. If a READY_IND isn't received by LEC B within a preset time-out value (C28) then LEC B may send a READY_QUERY message to LEC A asking for a READY_IND, or Data. The use of this time-out is optional.

Figure 8. UNI / PNNI / LANE Connection Signaling Example

10. Flush Protocol

A client has an option to send data frames to another client, via a broadcast VCC, before actually ever setting up or receiving confirmation of an established data direct connection to a destination client. If a client does this then there is a possibility that during the switch over from the old (possibly broadcast) path to the new path that frames may be received out of order. Other scenarios exist where an originator may have to switch paths for data delivery to a destination which can result in out-of-order delivery. To fix this an optional protocol called the "FLUSH message protocol" is provided.

The originator of the data frames, after sending data frames to the BUS, or data direct VCC, and after receiving confirmation that an alternate connection to the destination is established sends a LE_FLUSH_REQUEST message over the old connection, stores or discards future frames until (1) a

LE_FLUSH_RESPONSE to be returned via a Control Direct (or Distribute) VCC from its LES, or (2) a FLUSH time-out limit is reached. Only then will it begin sending data frames over the new path.

Transmitted LE_FLUSH_REQUESTs are each tagged with a Transaction ID. Each LE_FLUSH_REQUEST's Transaction ID is checked for in received LE_FLUSH_RESPONSEs. Note that LE_FLUSH_REQUESTs are not used for multicast frames; In-order frame delivery is not guaranteed.

11. Parameters that are Used by the LEC or LES During the LUNI Connectivity Process

Some configuration parameters must be known by a LEC before trying to join an ELAN. LANE Services can supply parameters during various phases of the connectivity process. There are 50 LEC parameters defined within the LUNI and LNNI standards. A summarized parameter overview is provided below.

1. LEC ATM addresses (specified as primary non-multiplexed, other non-multiplexed, or LLC multiplexed). A "primary" LEC address is used to connect to the LES and BUS, via CONTDIR, and MULDIR VCCs respectively. Other ATM addresses associated with a LEC are used for Data Direct VCCs. A single address can be limited to either a non-multiplexed or LLC-multiplexed VCC capable address. An address can also be specified as being capable of having both non-multiplexed, and LLC multiplexed VCCs.
2. LES address (unless statically configured, is given to a LEC by a LECS).
3. A Layer 3 address that is used in the Layer 3 TLV during connectivity process of a LEC.
4. LAN type that a LEC is, or wishes to become a member of (802.3 or 802.5).
5. AAL5-SDU frame size for Multicast and Data VCCs (maximum associated with non-multiplexed, and LLC multiplexed VCCs).
6. ELAN a LEC wishes to join.
7. LECID (assigned by the LES, if given permission to join an ELAN).
8. Local unicast and multicast MAC addresses associated with the LEC (to be registered with the LES).
9. Remote unicast addresses (these are not registered with the LES and are used in a Proxy LEC for responding to LE_ARPs).
10. Various capability Flags for:
    - Proxy capability (has remote unicast MAC addresses that are not registered with the LES).
    - LUNI V2, or V1 capable.
    - Selective Multicast capable.
    - LLC capable.
11. Various time-out periods for:
    - Control requests and responses.
    - Non-multiplexed data direct VCC (inactivity).
    - LE_ARP cache information.
    - FLUSH, and LE_ARP response.

- MULFOR setup.

12. Various delays for reconfiguration attempts or path switching if bypassing the FLUSH process, etc.
13. Various QOS parameters.
14. 802.5 parameters
15. Other parameters.

LES parameters: 9 parameters are defined. These must be known by an operational LES.

1. LES ATM address.
2. LAN Type (802.3 or 802.5).
3. Data frame size (maximum and minimum for LES and LEC AAL5-SDUs).
4. Control request and response time-out period.
5. Maximum frame age.
6. At least one BUS ATM address associated with the LES.
7. ELAN-ID.
8. 802.5 (Segment ID).
9. BUS Multicast Send VCC inactivity time-out period before LES termination of a LEC's ELAN membership.

E. LNNI V2

LUNI defines a set of LANE services as seen by LECs. LNNI provides a means to distribute LANE services and make them more reliable. LNNI defines the interaction of LANE services, not the interaction of services with LANE clients. Distributed services means that there can be multiple LESs, LECSs, BUSs, and SMSs serving a single Emulated LAN (ELAN). The interconnection of these services along with standardized line monitoring, configuration, and cache synchronization processes provides greater reliability than that offered by LANE v1. A LECS, however is the only service that can serve more than one ELAN. LNNI services are interconnected with Virtual Channel Connections (VCCs) for configuration, status, database synchronization, and control and message frame forwarding.

1. LNNI v2 VCCs and Components

In Figure 4, two dashed lines (labeled LUNI VCCs) cut through all associated LUNI v2 VCCs. The VCCs in the center of Figure 4 that are not cut by the dashed LUNI lines are defined and operate according to the LNNI v2 specification. The subset of LANE v2 VCCs in Figure 4 specific to LNNI are shown in Figure 9. Although some LUNI and LNNI VCCs share the same name (Configuration Direct, and Multicast Forward), they operate differently. The definition and operation of LNNI v2, and associated VCCs are the topic of this section.

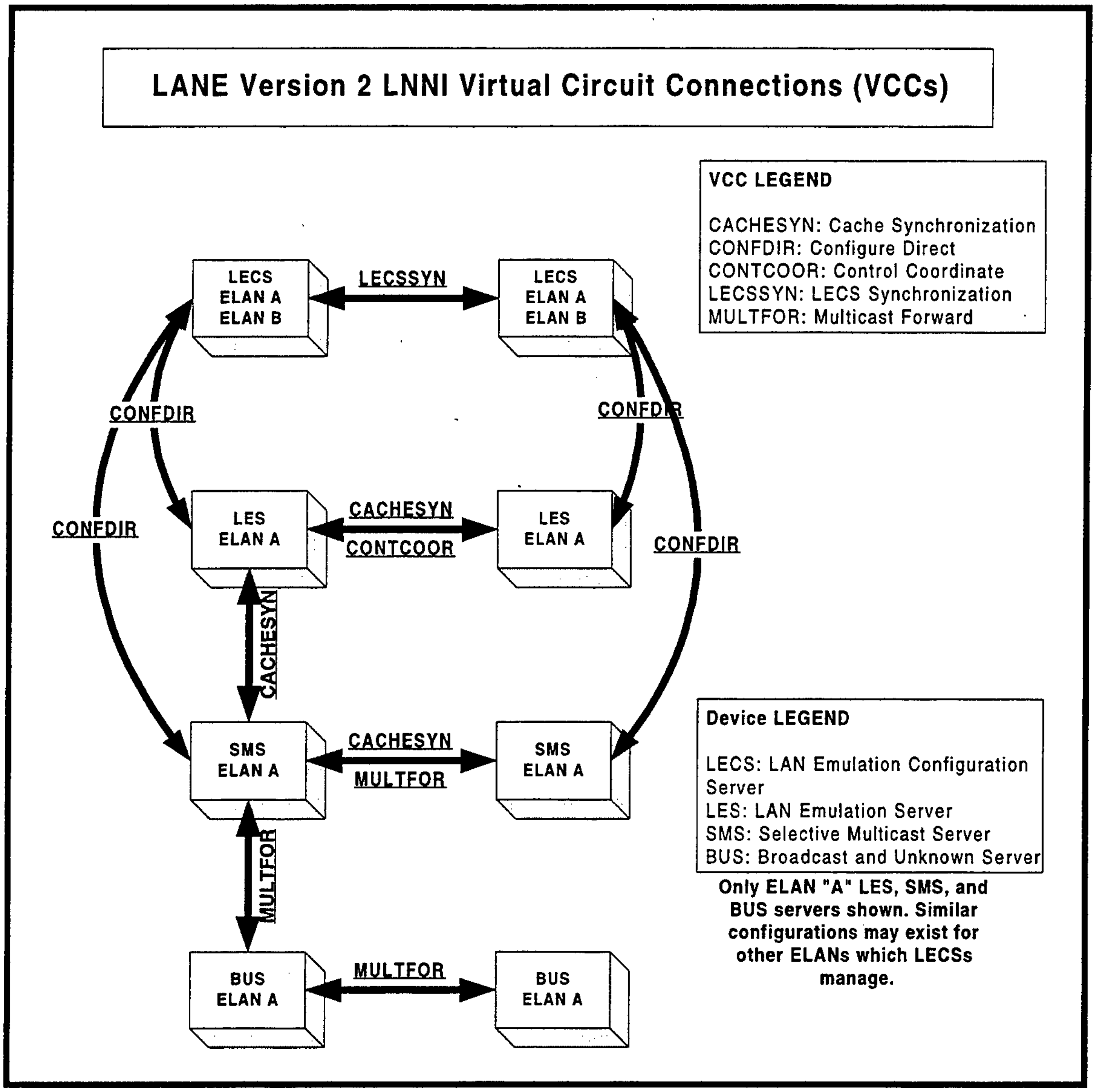

LANE Version 2 LNNI Virtual Circuit Connections (VCCs)
VCC LEGEND
CACHESYN: Cache Synchronization
CONFDIR: Configure Direct
CONTCOOR: Control Coordinate
LECSSYN: LECS Synchronization
MULTFOR: Multicast Forward
LECS ELAN A ELAN B
LECSSYN
LECS ELAN A ELAN B
CONFDIR
CONFDIR
CONFDIR
CONFDIR
LES ELAN A
CACHESYN
CONTCOOR
LES ELAN A
CACHESYN
SMS ELAN A
CACHESYN
MULTFOR
SMS ELAN A
MULTFOR
BUS ELAN A
MULTFOR
BUS ELAN A
Device LEGEND
LECS: LAN Emulation Configuration Server
LES: LAN Emulation Server
SMS: Selective Multicast Server
BUS: Broadcast and Unknown Server
Only ELAN "A" LES, SMS, and BUS servers shown. Similar configurations may exist for other ELANs which LECSs manage.

Figure 9. LNNI Specific Virtual Circuit Connections

A LECS primary function for LNNI is to keep configuration data for LESs, and SMSs, as well as to keep up to date records of which LESs, and SMSs are operational on a network. LNNI improves LANE v1 LECS service reliability, and performance by enabling a synchronized distributed LECS architecture. This means that there can be more than one LECS serving a network, and the information on every LECS is synchronized to be the same.

LECS are interconnected via a full mesh of LECS Synchronization (LECSSYN) VCCs. These VCCs can either be point-to-point, or point-to-multipoint, and are monitored via periodic keep-alive messages. Each LECSSYN VCC is setup using non-multiplexed Configuration Direct BLLI code-points in a UNI SETUP signaling message (See Figure 19, and the BLLI "User Information Layer 3 Protocol" field in Figure 20).

LECS are either preconfigured with the addresses of other LECS within a network domain or discover other LECS addresses on the network via ILMI. The LNNI v2 standard does allow for LECS to authenticate new LECS as peers before accepting them as peers, but does not mandate a means other than suggesting that they compare far end ATM addresses of connecting LECS to a preconfigured LECS ATM address list.

Each LECS is given the responsibility of reporting to peer LECS the operational state and associated parameters of all active LESs and SMSs to which it directly connects. A LECS Synchronization "keepalivetime" variable, and keep-alive process mandates that LECS Synchronization frames, which contain a list of all locally attached servers, must be transmitted before one-third of an agreed "keepalivetime" value. There are no consequences for the "keepalivetime" value being exceeded except that the LES status information within the LECS database is timed out. The result is that no new LEC/LES assignments would correspond to timed-out LESs. When LECSSYN VCCs are lost LECS continue to try to reestablish connectivity.

A LECS Synchronization frame contains a number of fields; however, other than operational server TLV fields, only the SOURCE-ATM-ADDRESS field is a variable that a transmitting LECS must have to know before sending a LECS Synchronization frame. The suggested LECS authentication mechanism would use this to determine if the LECSSYN setup message is from an authorized LECS; All of the other fields, except TLVs have standardized settings.

LANE can support the simultaneous operation of many ELANs within a domain. Each ELAN has its own set of LES(s), BUS(s) and/or SMS(s) that do not have to all connect to the same LECS.

Every operational LES and SMS sets up a CONFDIR VCC to a LECS. A CONFDIR VCC is used by a LES, and SMS to: (1) acquire configuration information (including, but not limited to Server IDs, and neighboring LES, and SMS server addresses) from a LECS, (2) receive or send server status information, and (3) monitor the LECS status (also used by LECS to monitor LECS and SMS status).

All LESs and SMSs are given a neighbor list (a list of LESs and SMSs that are directly attached in the same ELAN, and which must synchronize their caches), by a LECS. Each LES or SMS in a neighborhood list connects to each other via Cache Synchronous LLC-multiplexed VCCs to synchronize their registration databases using SCSP. Although each LES and SMS may only connect to a neighboring LESs and SMSs with Cache Synchronous (CACHESYN) VCCs, the union of all LES and SMS neighborhood lists and corresponding CACHESYN VCCs extends connectivity to every LES and SMS in each ELAN. After LESs cache synchronize (via SCSP), the following occurs:

1. Each LES sets up Control Coordinate LLC-multiplexed VCCs in a full mesh to all other LESs (not just its neighbors) in its ELAN. Control Coordinate VCCs are used to forward control messages such as LE_ARPs.
2. Each BUS (logically paired and on the same physical device with a LES) sets up non-multiplexed multicast forward VCCs in a full mesh to all other BUSs in its ELAN.

Server VCC acceptance and rejection rules are provided below. These rules may cause duplicate VCCs to be aged out due to inactivity without the possibility of a race condition removing too many VCCs.

> A server MUST accept a UNI signaling request to establish an LLC-multiplexed VCC to its ServerMuxedAtmAddress, with an assumption that this VCC will be used as a Cache Synchronization or Control Coordinate VCC. ... The server MAY release this connection if, upon completion, no valid synchronization or control communications use the VCC within the IdleLaneControlVccTimeout, AND the calling party address does not match a configured server address in its SynchronizationPeerServerList. ...
>
> When a server receives an incoming Cache Synchronization connection request from an ATM address to which it already has a connection, it should accept the request, and apply the rules for duplicate Data Direct VCCs found in Section 8.1.15 of LUNI v2 for aging out the appropriate duplicate VCC." Note that this does not imply the allowance of duplicate address registrations; It only allows duplicate connections from a preregistered address. ...
>
> When an LE Client is told to send a data frame via LE_UNITDATA.request, it should use a VCC matching the QOS requested for that frame. If multiple matching VCCs exist, the LE Client MUST use the VCC whose calling party ATM address is numerically lower than the called party address. If multiple such VCCs exist, the VCC which was the earliest created (That is, the VCC whose SETUP or CONNECT message was received first) MUST be used. [5]

Each LES keeps a distributed synchronized registration database (LNNI database) of all MAC address, ATM address, Server IDs, and other operating parameters of every member of an ELAN. The SCSP process synchronizes LNNI database information on LESs as well as between SMSs within each ELAN. When a LES receives indication that a LES is no longer functioning all LNNI databases remove information pertaining to LECs associated with the non-operational LES.

BUSs are paired with LESs. LES/BUS pairs are physically and logically co-located on a single device. Pairing of LESs and BUSs assumes that each BUS has full access to paired LES registration database information; for this reason intercommunication specifications are not defined between LESs, and BUSs in LNNI v2 [5].

BUSs are sent broadcast, multicast, and unicast traffic that have unresolved ATM addresses. They forward this traffic to local LECs and to other BUSs on an ELAN. BUSs and neighboring SMSs are interconnected via MULTFOR VCCs so that, if required, selective multicast traffic received by a SMS can be relayed to LANE v1 clients, which only have multicast forward connections from local BUSs.

SMSs are interconnected with neighboring LESs and other SMSs via CACHESYN VCCs (using SCSP) so that they can maintain a table of all local LECs that are registered with the LES as receiving, sending, or both receiving and sending multicast traffic for specified multicast addresses by an assigned SMS. SMSs are used to offload selective multicast traffic from BUSs. SMSs learn which clients require multicast forward VCCs after notification of newly registered and assigned clients via cache synchronization with a neighboring LES.

2. Server Cache Synchronous Protocol (SCSP)

SCSP (RFC 2334) provides a way for a distributed set of servers to synchronize the information they have about the clients that they serve. Information is grouped by Protocol ID (PID) and Server Group ID (SGID), which when paired define a single instance of the SCSP protocol. Multiple instances of SCSP can coexist on one network.

There are three phases to the SCSP: (1) Hello phase, (2) Database Synchronization phase, and (3) a Flooding phase. The Hello phase involves a server determining and monitoring the operational state of neighbor servers and associated connectivity. After a connection is made to a potential server, a Hello message is sent on a newly formed CACHESYN VCC (which may be a point-to-multipoint VCC) to the neighbor server to discern the operational state of the neighbor server. The SCSP protocol monitors each SCSP connection to neighbor servers to determine if they are non-functional, uni-directional, or bi-directional. Only after bi-directional connectivity is ascertained are Hello messages sent.

A list of potential neighbor servers and their SID values can be pre-configured within a server, discovered by a LECS, or also learned via received Hello messages. This is a very trusting protocol. A "Hello" protocol is defined within the SCSP standard to monitor neighbor connections. Once a neighbor server connection is up, there is an exchange of Hello messages. These Hello messages include the originating Server ID (SID), all of the its presumed neighbor SIDs, a Hello Interval (time between the sending of Hello messages), and a Dead Factor (a Hello Interval multiplier used for fault recovery). After an exchange of Hello messages, the neighbor servers enter a Data Synchronization phase.

Database Synchronization and Flooding phases use a "Cache Alignment" (CA) protocol and a "Cache State Update" (CSU) protocol defined within the SCSP specification. The Cache Alignment protocol synchronizes caches between two servers, and the CSU protocol is used to update caches on neighboring servers.

In order to clarify the remaining SCSP discussion, terminology defined within the SCSP specification is used. SCSP defines a Local Station (LS), a Directly Connected Station (DCS), as well as a Remotely Connected Station (RCS). The following SCSP description uses the same terms as the SCSP standard. A LS could be a LES, which receives a neighbor list of DCSs from a LECS. A RCS is another LES in the same ELAN, but not a neighbor of the LS.

For the purposes of discussion consider a LS as a machine which has just booted, discovered its neighbors, setup CACHESYN VCCs, and finished the Hello phase. The first CA message sent contains a CA header, Local Station ID (LSID), a DCS SID list, three flag bits, and no Cache State Alignment Summary (CSAS)s. Each server negotiates cache control uses the three flag bits in the CA message. The DCS SID receives this information, sends back a similar CA message, and then begins the Database Synchronization phase of the SCSP.

The Database Synchronization phase starts with a Cache Summarization process. The Cache Summarization process starts with each server sending Cache Alignment (CA) messages that contain zero or more records. In the scenario noted above, a LS would receive these CSAS records from the DCS, process them, and create a list of Cache State Alignment (CSA) records that it needs called a CSA Request List (CRL).

The LS uses the CRL to create Cache State Update Solicit (CSUS) messages. CSUS messages contain the summarized CSAS records that the LS needs the DCS to send to it. The LS sends the CSUS messages to the DCS. Upon receiving the CSUS the DCS sends the requested record information to the LS via CSU request messages. The LS replies back to the DCS with CSU reply messages. Once all of the messages are sent the caches are aligned, and the Flooding phase begins.

The Flooding phase is actually a repeat of the Cache Alignment process, but is extended between DCSs and RCS, with each DCS taking on the role of a LS, and each RCS taking on the role of the LS's DCS. This is done to propagate new or changed cache information to all servers within a SCSP instance. This process involves each server sending CSU requests containing the newly learned information to all DCSs (except possibly the one in which the CSA change was learned), and so on, until all server caches are aligned. Note that CSU requests are followed by associated CSU replies to acknowledge request reception. Information is resent, if unacknowledged.

The SCSP protocol optionally supports server authentication, via SCSP packet extensions. Vendor-Private extensions can also exist. Note the following security related statement.

> If authentication extension is not used, or if the security is compromised, then SCSP servers are liable to both spoofing attacks, active attacks, and passive attacks. ... Any SCSP server is susceptible to Denial of Service (DOS) attacks. A rouge host can inundate its neighboring SCSP server with SCSP packets. ... If security of any SCSP server is compromised, the entire database becomes vulnerable to corruption originating from the compromised server. [6]

F. LNNI ASSOCIATED FRAME FORMATS

The frame formats for LE (LUNI and LNNI) Control frames are shown in Figure 5. OP-CODE values distinguish control frame message types. LNNI control frames are LLC-multiplexed with a prepended LLC header shown in Figure 6 (note the Frame Type field). Each SCSP message contains three parts: (1) a fixed part, (2) a mandatory part, and (3) extensions part.

An example SCSP message showing the fixed and mandatory parts for a SCSP "Hello message" is shown in Figure 10. Note that the mandatory part is different for each message type. The Extensions part is not shown, but it contains one or more Type, Length, Value (TLV) grouping for each extension type. Two Extension types are defined in SCSP, a SCSP Authentication extension and a SCSP Vendor-Private extension. Authentication methods are not standardized. Vendor-Private extensions can vary.

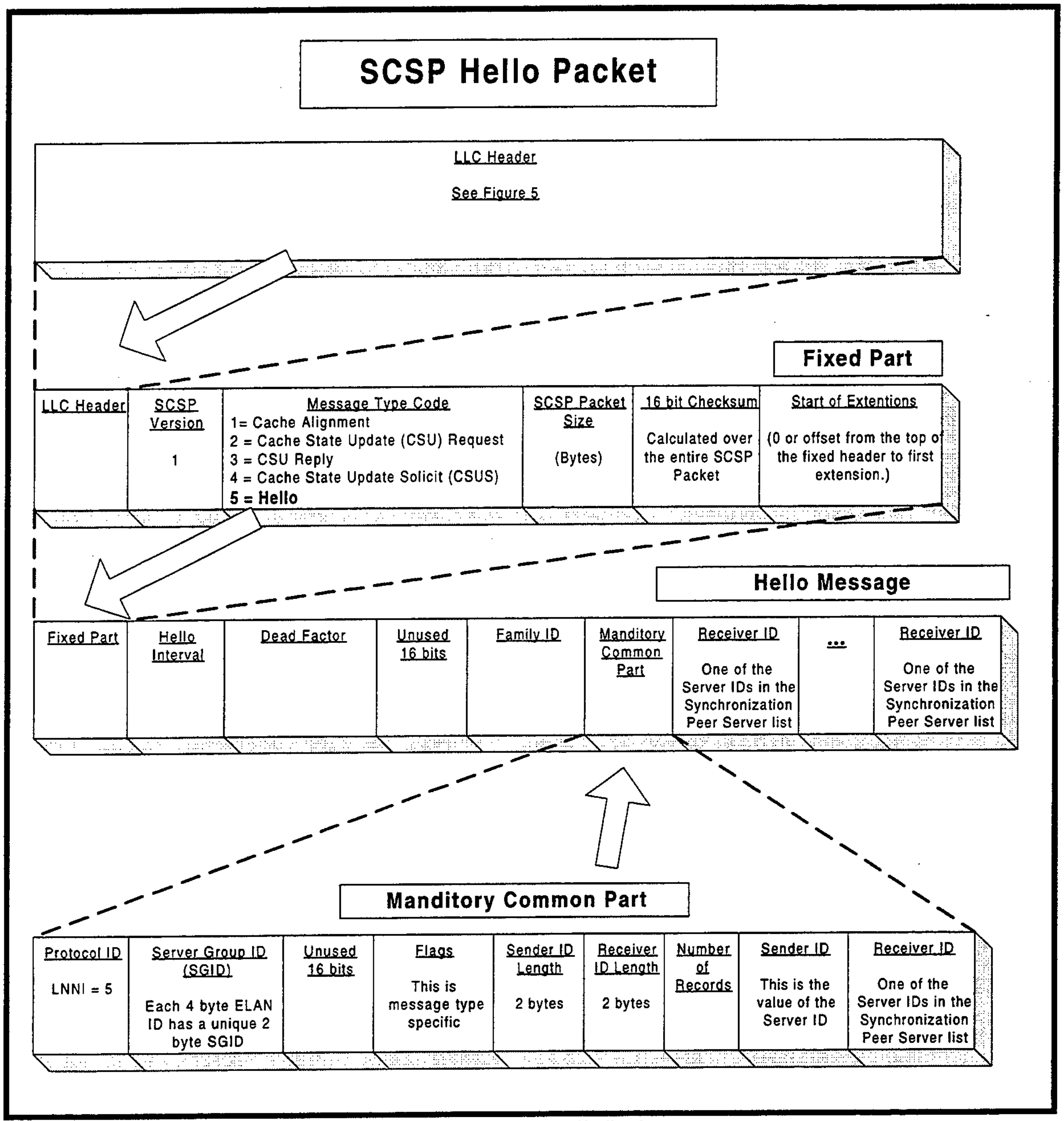


Figure 10. SCSP Hello Packet

G. LAN EMULATION CLIENT MANAGEMENT SPECIFICATION VERSION 2

The LAN Emulation Client Management Specification v2 (LECMv2) provides a means to configure and manage LECs in a network. The LAN Emulation Servers Management Specification Version 1 (LESMv1) in Section H of this Chapter provides network management to LECS, LES, and BUS servers in an ELAN. Both the LECMv2 and LESMv1 address configuration, performance, and fault management. A high level overview of the LECMv2 is the topic of this section.

LECMv2 is based on the Simple Network Management Protocol Version 2 (SNMP v2). It requires that systems which support SNMP implement RFC 1213 (MIB-II), RFC 1573 (Evolution of MIB-II), RFC 1695

(Definitions of Managed Objects for ATM Management – called AToM MIB), as well as the LECMv2 itself. Note that additional MIBs may also reside on a host, such as vendor specific MIBs or Bridge MIBs.

LEC Management MIB groups defined in LECMv2 contain objects which pertain to all initial state parameters, operational state parameters, performance, and connection management variables associated with a LEC. Each group is row indexed just like MIB-II/RFC 1573 to correspond to a single client. A mapping between MIB-II/RFC 1573, and LECMv2 row indices is provided within the LECMv2 MIB. A total of 13 groups defined in this specification are as follows:

1. Client Configuration Group – Mostly initial configuration parameters.
2. Client Status group – Some initial configuration parameters as well as others updated during Configuration or Join phases of ELAN connectivity.
3. Client Statistics Group – Performance statistics.
4. Client Server Connections Group – The Interfaces and VPCIs associated with the servers a LEC is connected to.
5. ATM Addresses group – A list of ATM addresses associated with a LEC.
6. Registered LAN Destinations MAC Addresses group – MAC address and binding information to ATM addresses.
7. Registered LAN Destinations Route Descriptors group - (for token ring)
8. LE_ARP cache group for MAC Address translations – LE_ARP cache store for LEC operations.
9. LE_ARP cache group for Route Descriptor translations - (for token ring)
10. Index Mapping group – A mapping of MIB-II/RFC 1573 interface index to/from LEC indexes.
11. Multicast Forward VCC group –VPCI values of Multicast Forward VCCs.
12. Proxy LE_ARP response group – Contains LE_ARP response information associated with proxied clients.
13. TLV group – TLV grouped information.

All MIBs are written in the standard Abstract Syntax Notation 1 (ASN.1). They define not only each object instance, but also configurable permissions, and values for each. The MIBs are compiled and put on each device managed.

H. LAN EMULATION SERVERS MANAGEMENT SPECIFICATION VERSION 1

LESMv1 defines the configuration and management information for LECS, LES, and BUS servers. The configuration and management information is divided into three MIB modules: (1) an ELAN MIB module for a LECS, (2) a LES MIB module for a LES, and (3) a BUS MIB module for a BUS. MIB modules pertaining to other protocols, such as the AToM MIB also reside on LANE servers. As with LECMv2, the LESMv1 module also is based on SNMP and ATM MIB RFCs and specifications.

The ELAN MIB deals with the information a LECS uses to decide which ELAN a client can join (policies), and the information the LECS needs to provide to a client to join an ELAN. It also deals with the configuration, performance, and fault management of the LECS itself. The following is a summary of the ELAN MIB groups, and generalized functions:

1. ELAN Administration group – Provides a registry of ELAN assignment policies. Suggested ELAN assignment policies could be based on one or more of the following: ATM address, MAC address, Route Descriptor, LAN type, packet size, or ELAN name.
2. ELAN Configuration group – Provides an ELAN configuration table with associated LES table, Policy table, and various assignment tables.
3. LECS Configuration group – Provides LECS configuration and monitoring parameters.
4. LECS Statistics group – Provides LECS counter variables associated with LECS statistics.
5. LECS Fault Management group – Provides error-logging functions.

The LES MIB deals with LES configuration, performance and fault management associated with the LES itself, and various connections to it. The following is a summary of the LES MIB groups, and generalized functions:

1. LES Configuration group – Provides LES configuration parameters such as ATM address, ELAN name etc., as well as VCCs with links to the AToM MIB VCC indices, LE_ARP table, and a LES-LEC topology table.
2. LES Statistics group – Provides performance and fault counters.
3. LES-LEC Statistics group – Provides LES-LEC related error statistics
4. LES fault Management group – Provides an error logging capability.

The BUS MIB is very similar with the LES MIB in that it provides for the configuration, statistics, and fault management of the bus. This is not described any further.

IV. ATM

The ATM model (See Figure 11) is represented in its most simplistic form as a three dimensional cube consisting of horizontal layers and vertical planes. Figure 11 references four acronyms: ILMI (Interim Local Management Interface), UNI (User Network Interface), SVC (Switched Virtual Circuit), and PVC (Permanent Virtual Circuit). Each plane and layer has specific functions and work together.

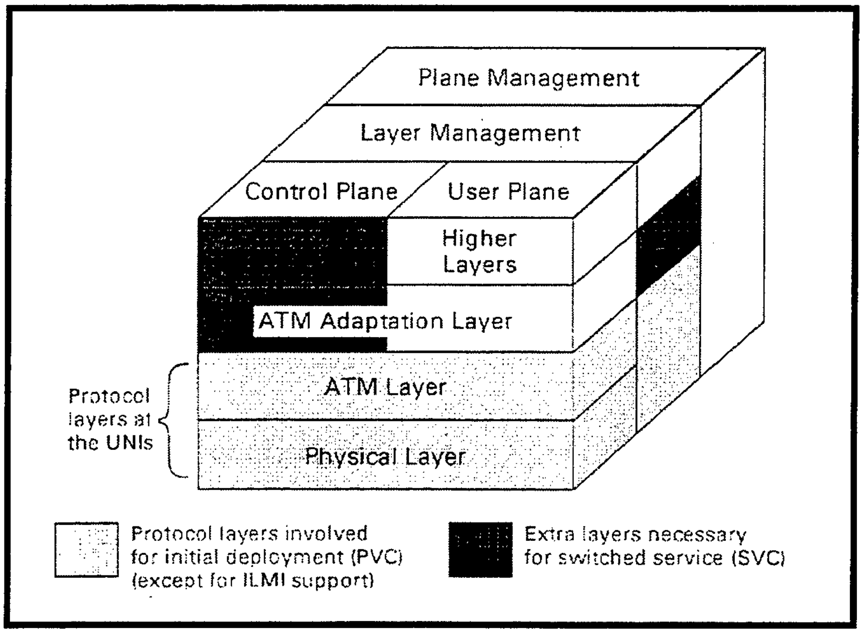

Plane Management
Layer Management
Control Plane
User Plane
Higher Layers
ATM Adaptation Layer
ATM Layer
Physical Layer
Protocol layers at the UNIs
Protocol layers involved for initial deployment (PVC) (except for ILMI support)
Extra layers necessary for switched service (SVC)

Figure 11. The ATM model [7]

The Physical layer and an ATM layer are common to all three planes (User, Control, and Management). The Physical layer provides an interface and signal conversion to transport and receive ATM cells across a physical medium. It also generates and verifies Header Error Check (HEC) field contents on transmitted and received cells. NetX uses a fiber based physical medium with SONET/Synchronous Digital Hierarchy (SDH) protocols. The ATM layer provides upper layer cell identification and a cell header, which includes a field for physical layer HEC placement.

The User Plane (U-Plane) provides for the transfer of user application information. It consists of the Physical Layer, ATM Layer, and multiple ATM Adaptation Layers (AAL), and higher layers. Note, Figure 11 shows one AAL layer, but this layer may be one of several different types (application dependent).

The Control Plane (C-Plane) provides call establishment and release and other connection control functions necessary for providing switched services. This plane has AAL procedures; higher layer signaling protocols, and shares the Physical and ATM Layers with the U-Plane.

The Management Plane (M-Plane) provides management functions, and the capability to exchange information between the U-Plane, and C-Plane. The M-Plane contains two management entities, the Plane Management entity, and the Layer Management Entity. The functions of the Plane and Layer Management Entities are just what their names suggest, management of the Planes, and Layers, respectively.

A. PHYSICAL LAYER

The Physical Layer is divided into two sublayers, the Transmission Convergence (TC), and the Physical Medium (PM) sublayer. NetX uses SONET/SDH as a transport mechanism over optical fiber. A detailed view of a SONET/SDH frame, called a SONET frame in this thesis, as defined by UNI3.1 is shown in Figure 12. Each STS-3c frame is made up of 270 columns by nine rows of byte size blocks. Each frame includes a Transport Over Head (TOH – first 9 columns) and a Synchronous Payload Envelope (SPE – remaining 261 columns). The TOH is composed of a Section Over Head (SOH – first three rows), and Line Over Head (LOH – remaining 6 rows). The SPE includes a Path Over Head (POH – first column of the SPE), and information bytes. Basically the SOH is accessed at each end of a physical cable. The LOH is accessed whenever a frame is terminated, or multiplexed into a higher or lower STS rate signal. The POH is only accessed when the SPE is terminated. Note that if a STS signal is terminated and demultiplexed into DS3 signalls it is considered a path termination and line termination. This is because the multiplex process terminates the SONET path information, and does not just multiplex the frame to a higher or lower rate STS type transmission paths.

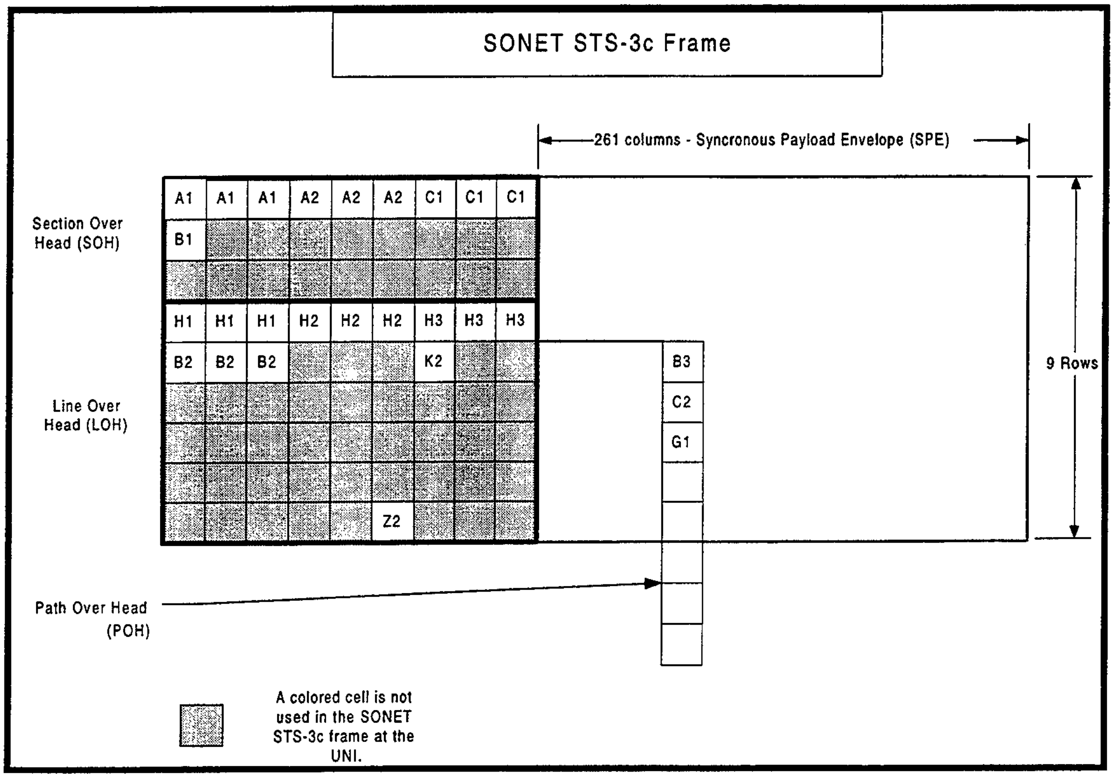


Figure 12. A detailed view of the SONET STS-3c Frame

The physical layer provides Operations and Maintenance (OAM) correspondence via Section, Line, and Path overhead bytes in a SONET frame. The Physical layer can insert "idle" cells into the SPE of the SONET frame if it does not receive enough "user" cells to fill a SPE; these cells have special header values that are recognizable by the physical layer. It is specified in I.361 that the physical layer can insert "OAM" and "reserved" cells into the SPE, as well as the "idle" cells. Note that OAM cells or other types of "Reserved" cells are for future use, or deal with other types of transmission mechanisms besides SDH (e.g., Cell-based transmission defined in I.432). In the SONET/SDH case, the ATM layer (not the physical layer) provides OAM cell placement into the SPE.

When ATM cells are sent from the ATM layer to the Physical layer a HEC calculation is performed on the cell header and the HEC results are placed into byte 5 of the ATM cell header. Several Bit-Interleave-Parity calculations are performed, every 48-byte cell payload is scrambled, pointers are established in SONET frame overheads, and the cells are inserted into the SPE. Then the entire frame is scrambled except for the first row of the SOH, and sent out of the physical interface. The reverse is true in the opposite direction.

The UNI 3.1 standard requires that all equipment supporting the UNI must ignore all overhead bytes/bits undefined at the UNI (including the SDH defined Data Communications Channels). It is also a requirement of the UNI 3.1 standard that all undefined overhead bytes (colored blocks in Figure 12) be encoded as zero before they are scrambled and transmitted.

1. Detailed Physical Layer Functions

The Physical Layer performs many functions. A listing of those functions with comments are noted below:

1. ATM HEC generation/verification (see Figure 16) - The entire header (including the HEC byte) is protected by a Header Error Control sequence. Two modes are available, Correction mode, and Detection mode. Correction mode will correct all cell headers with single bit errors, and discard cells with headers that have multiple bit errors. Detection mode will discard any cell that has a header error. The HEC field consists of the remainder of the Generator Polynomial $x^8 + x^2 + x + 1$ divided into the content of header multiplied by x^8. The HEC check process detects 100 percent of single bit errors, but only 87% of multiple bit errors [8]. Note that if a bit error should occur in the transmission medium the frame descrambling process will produce multiple bit errors.
2. Cell payload scrambling and descrambling - A SDH based self-synchronizing scrambler with polynomial $1 + x^{43}$ is used to scramble each cell payload (not the header). The scrambler algorithms at each end of a link are synchronized with a 43-bit sequence in the first cell of the first frame at start-up, or a restart following a LOS. The scrambling provides a continuous variable payload bit pattern to improve the efficiency of the cell delineation algorithm, and preventing any of the frame contents from replicating the frame alignment bytes (A1 and A2).

3. Cell delineation - The first bytes of a frame (A1 and A2) are called frame alignment bytes. When these bytes are detected (Hex F628) a frame descrambling polynomial is reset, and all of the bytes following the first row of the SOH are descrambled. After this, a hunt state is entered where each bit is scanned until a pattern of bits reflects a correct HEC calculation on the previous four bytes. After this point a "presync" state is entered, and a cell by cell scan is made to determine if the correct HEC placement has been found. If all of the HECs are correct the device enters a "sync" state, where it know knows the correct cell boundaries for the rest of the frame. The process of finding cell boundaries is called "Cell Delineation."
4. Path signal identification (C2) - This provides a preset bit pattern for the C2 byte in the SONET overhead to identify the SONET payload frame type, i.e., 00010011 signifies a SONET Synchronous Transport Signal (STS)-3c payload.
5. Multiplexing-using VPI/VCI fields - The physical layer identifies four types of cells: (1) physical layer OAM cells, (2) idle cells, (3) errored cells, and (4) cells that are associated with higher layers. The first three types are not sent to higher layers. Cells that are cells with multiple bit errors in their headers and that are not detected by a HEC can still be filtered out if the corrupted header pertains to invalid VPI/VCI/PT combinations known to the physical layer.
6. Physical Frame Payload scrambling/descrambling - Each SONET element must have the capability to derive its clock timing from an incoming OC-N signal. Therefore, it is important to maintain ones density in a data stream. Scrambling using a polynomial 1 + X^6 + X^7 is used for this purpose. It is reset for each frame to a "1111111" and scrambles all of the frame except the first row of the SOH.
7. Cell rate decoupling - This provides the same function as the ATM layer Cell Rate Decoupling function. Physical layer "Idle cells" are inserted to provide the same function as the ATM layer's "unassigned cells."
8. Pointer processing - Used by SONET to locate smaller synchronous payloads within a SONET Synchronous Payload Envelope (SPE), i.e., an STS-1 within an STS-3C. Typically if the clock used to create the overhead is different from that used to create the payload a pointer to the first STS-1 is provided, with concatenation indicators used for the remaining STS-1s within an STS-3c.
9. Bit timing - Each SONET element derives its clock timing from an incoming OC-N signal. Frame scrambling helps facilitate this.
10. Transmission frame generation/recovery - This is the normal transmission of a frame on a physical medium and the recovery of such frames at a receiver.
11. Physical medium transmission specification - The physical medium transmission specification deals with the conversion and transmission of signals over a physical medium. Multimode, or single mode fiber optic cable (for NetX). The physical medium uses the (SONET/SDH) standards as defined in the ANSI, ITU, and the ATM Forum.

12. Bit Interleaved Parity checks - Each SONET frame contains a Section Bit Interleaved Parity (BIP8) byte (B1), three Line BIP8 bytes (B2), and a Path BIP8 byte (B3). The Section BIP8 (B1) is calculated over the entire previous STS-N frame after it is scrambled. The Line BIP8 (B2) is calculated over the previous STS-1 (except the first three rows of the Section Overhead) before scrambling (note that since three STS-1's are byte interleaved within a STS-3c payload there are three B2 bytes in the line overhead). The Path BIP8 (B3) is calculated over the previous STS-N SPE before it was scrambled. These BIP8 bytes are used to detect errors in transmission.
13. Monitoring, performance analysis, and fault management - The monitoring of the B1, B2, and B3 bits, input light pulse transitions, as well as AIS, FEBE, and RDI indicators provide a measure of the performance of a section, line, or path at the physical layer. The following problems, results, and outcomes of results are typical of a SONET/SDH interface used at a UNI.

2. An Example Showing Physical Layer Errors and Resulting Indications

Seven different error scenarios and resulting indications are provided in the list below.

1. No Input light pulses detected for 2.3 to 100 microseconds. - Results in a Loss of Signal (LOS) indication.
2. No framing bits (A1 and A2) were identified for 4 frame times. - Results in a SEF indication.
3. While in a Sync State, up to 7 HEC errors were detected. - Results in an Out of Cell Delineation (OCD) indication.
4. Pointer or New Data Flag (NDF) problems associated with the H bytes in the LOH persisted for 8 frame times. - Results in a Loss of Pointer (LOP) indication.
5. A Severely Errored frame (SEF) persisted for more than 3ms. - Results in a Loss of Frame (LOF) indication.
6. An Out of Cell Delineation (OCD) persisted for more than 4ms. - Results in a Loss of Cell Delineation (LCD).
7. B2 or B3 errors were detected. - A count of the number of errors with Line B2, or Path B3 is returned to the sender in the form of FEBE-L indications (using the Z2 byte), or FEBE-P indications (using the G1 status byte) for performance analysis and fault maintenance.

Resulting actions to the indications provided in the above list are dependent upon the device type receiving the indication. Section, Line and Path terminating equipment actions derived from ANSI [9] are as follows.

1. Section Terminating Equipment detects either a LOS or LOF it sends an **AIS-L** downstream. An AIS-L is a frame containing a valid SOH with the K2 byte set to 1 and a scrambled all 1's pattern for the rest of the signal. The AIS-L is used to not only indicate a signal defect, but also to provide clock recovery (sending a keep-alive) at downstream STE and LTEs.

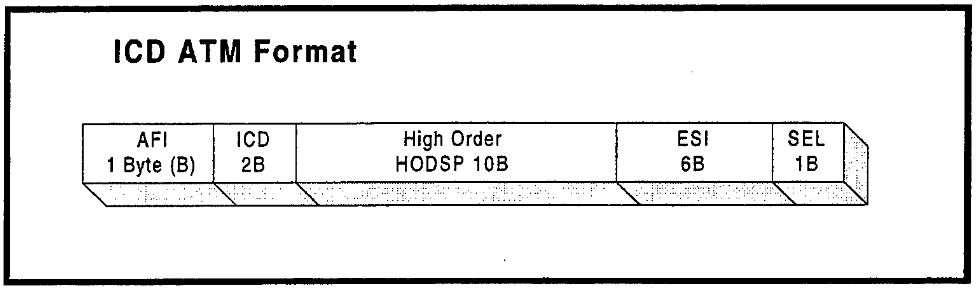
2. Line terminating Equipment (LTE) detects a LOS, LOF, LOP, or AIS-L it generates an **AIS-P** downstream to the associated PTE. An AIS-P is an all 1's in the H1, H2, H3, and SPE within the frame. If the LTE detects a LOS, LOF, or AIS-L it generates a Remote Defect Indication – Line (**RDI-L**) and sends it to the transmitting PTE that the incoming LTE has detected a problem with.
3. STS Path Terminating Equipment (PTE) detects LOS, LOF, LCD, LOP, AIS-L, it generates an RDI-Path (**RDI-P**) to the associated PTE that the incoming PTE has detected a problem with. If the PTE detects LOS, LOF, or AIS-L, it generates an **RDI-L** to the LTE that the PTE has detected a problem with.

The bottom diagram in Figure 16. Shows the order in which Bytes are ordered in a Physical layer SONET OC-3 frame. This frame is generated by an OC-3 (155 Mbps) fiber-optic interface every 125 microseconds. This OC-3 interface standard is used to connect Edge Devices with ATM backbone switches on the NetX backbone. Another frame, similar to this, but with more columns, and operating at 622 Mbps is generated by OC-12 interfaces between ATM switches on the NetX backbone. The functionality of the OC-3, and OC-12 frame formats are similar. Each contain: Section, Line, and Path overhead bytes for such things as frame bytes, pointers, alarm signals, and error monitoring bytes (parity bytes) for the SONET payload.

B. ATM LAYER

An ATM cell is identified by a VPI/VCI/PT field combination, and an ATM system is identified by a 20-byte ATM address. An ATM address is used to setup a connection (circuit) to an end system, upon which cell transmissions are identified by assigned VPI/VCI/PT field combinations.

Each logical node or physical device on an ATM network has a unique 20 Byte address. For NetX the address format is called the International Code Designator (ICD) format, and the registration authority for the ICD is the British Standards Institute. The format for an ICD address is shown in Figure 13.



ICD ATM Format

AFI 1 Byte (B)	ICD 2B	High Order HODSP 10B	ESI 6B	SEL 1B

Figure 13. An ATM Address – ICD Type

The acronyms in Figure 13 are: Authority Format Identifier (AFI), International Code Designator (ICD), End System Identifier (ESI), Selector Byte (SEL), and Byte (B). The entire address is 20 Bytes long.

The AFI and ICD basically pinpoint the authority under which the address is registered. NetX coding of the HODSP is defined by the ICD authority, and could be used to facilitate routing through interconnected ATM networks. The ESI is an 802.3 MAC address of a connecting system. The SEL Byte is only used by end systems to distinguish entities specific to that system (e.g. between PNNI hierarchical nodes on a single physical device, or between LECs on an ATM station. Note, that since NetX is not connected directly via a signaling ATM interface to other ATM networks (NetX connects to off-site networks via an Ethernet router – see Figure 1) the use of a registered address is not needed. To preplan for ATM signaling-based off-premise ATM connectivity in the future suggest that an addressing space for NetX be registered.

The ATM Layer at a high level performs cell multiplexing and payload type coding/recognition functions, as well as facilitating ATM layer F4 (segment), or F5 (end-to-end) OAM functions identified in ITU-T I.610, ITU-T I.371, and ITU-T Q.2120. ATM layer F4 or F5 functions provide alarm surveillance, and connectivity verification. The ATM layer also provides traffic, and congestion control functions, independent of upper layers. It only operates only on pre-established ATM connections, and uses three fields, VPI (Virtual Path Identifier), VCI (Virtual Channel Identifier), and PT (Payload Type) to identify cell types for multiplexing. ATM layer functions (excluding F4 and F5 management functions) are as follows:

1. Adds/Removes a 5 byte ATM header. - This header includes a Generic Flow Control (GFC) field, VPI and VCI fields, a Payload Type (PT) field, and a Header Error Control (HEC) field.
2. Discriminates between ATM user, control, operation, administration and maintenance (OAM) cells. - The discrimination is based upon the information content of the VPI, VCI, and PT fields. Any cells that the ATM layer receives from the Physical layer that are not associated with an established circuit, or functions with pre-defined (VPI/VCI/PT) combinations are discarded.
3. Multiplexes among different ATM connections. - The multiplexing function is related to maintaining a QOS associated with the connections established at setup time.
4. Allows the setting of cell loss priority. - This is a one-bit field that any ATM device can set in a cell header to maintain a distinction of high and low priority cells. A cell that has the CLP bit set to "one" is a low priority cell that can be discarded in the event of unexpected congestion.
5. Adds "unassigned cells" to support the synchronous cell time slots of the physical layer (when needed). - This is referred to as Cell Rate Decoupling. It is used to transform a non-continuous cell stream of assigned cells into a continuous stream of assigned and unassigned cells. This rate is determined by the physical layer transmission rate. These unassigned cells are recognized by specific header patterns. SONET requires cell rate decoupling, because a continuous flow of cells must be sent.
6. Optionally provides for traffic shaping. - This is used at ATM end points to provide a mechanism to control the speed at which cells are sent by a device in order to meet an established QOS. Examples are Peak Cell Rate reduction, and burst length limiting.

Section 3.5 in the UNI 3.1 standard discusses the ATM Layer Management plane (M-plane) specifically with regard to the functions required by a private, or public UNI interface. A subset of functions defined in ITU-T Recommendation I.610 and ANSI T1S1.5/92-029R3 are used and specified in Section 3.5.3 of the UNI 3.1 standard. Those functions are associated with alarm surveillance of VPs, connectivity verification of VPs, and VCs, and the detection of invalid VPI/VCI/PT combinations.

The ATM layer M-plane performs traffic and congestion control, independent of any upper layer. It also provides traffic, and congestion information to upper layer applications that perform application level traffic, and congestion control. Note that application level traffic control, and congestion mechanisms are independent of any ATM layer congestion, and traffic control means.

Analogous to the physical layer F2, and F3 management functions, the ATM layer provides F4 (VP) and F5 (VC) management at the ATM Layer (See Figure 14). Both F4 and F5 functions can be used for segment, or end-to-end management. End-to-end OAM cells are carried transparently through intermediate ATM switches unless otherwise configured. Note that any OAM cell that is allowed to pass through an ATM node may be monitored for content. Segment OAM functions as specified in the UNI 3.1 standard operate on single VP, or VC links between two ends of a UNI link. Public UNI interfaces can block F4, and /or F5 OAM flows.

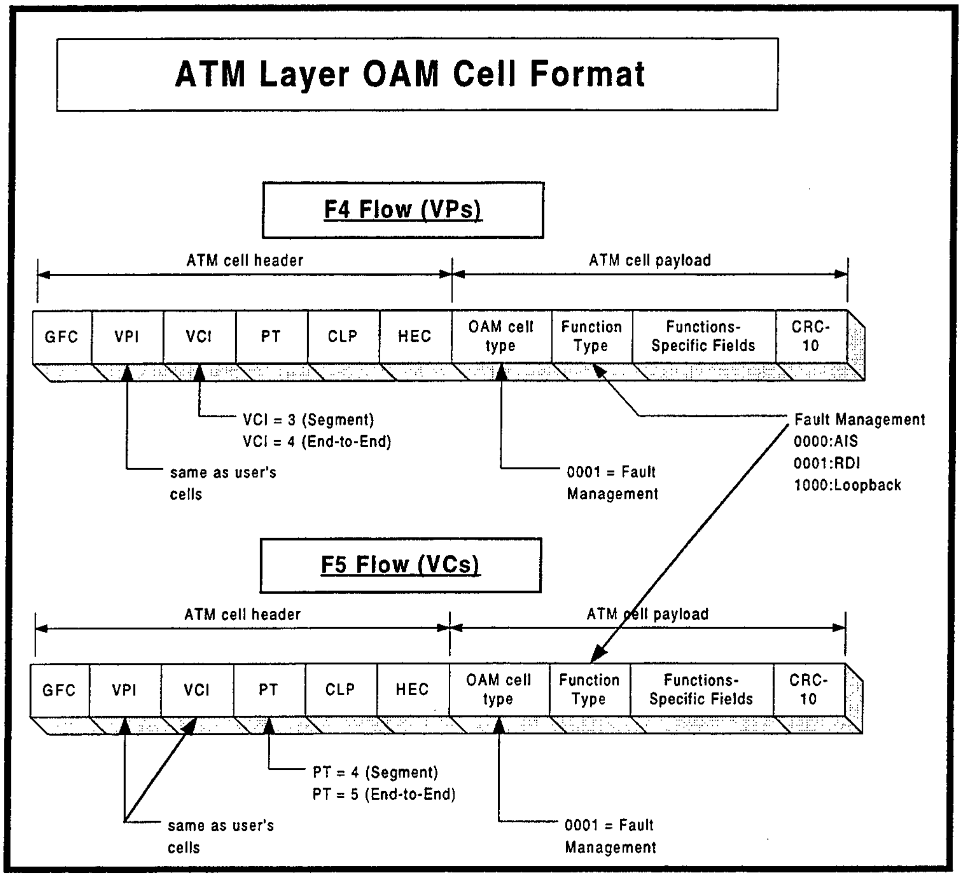


Figure 14. ATM Layer OAM Cell Format

Alarm surveillance at a public UNI detects, generates, and propagates failures via VP-AIS/VC-AIS, VP-RDI, and VC-RDI failure indications in a F4, or F5 OAM cell. The AIS and RDI indications (identified within the Function Type field in an OAM cell) are analogous to the physical layer AIS and RDI indications. In the event of failures, AIS indications are sent downstream, and RDI indications are sent upstream. The VP-AIS and VC-AIS failure indications can be caused not only by a detection of a VPC/VCC failure, but also by notification of a physical layer failure. The receipt of a VP, or VC AIS triggers the generation of associated VP, or VC RDIs. Note that these indications are required for Public UNIs, not Private UNIs.

VP and VC connectivity verification is provided by "Loopback" OAM cells that when received are required to be looped back to the sender within one second. There is no requirement for the ATM layer OAM Loopback cell capability to support delay measurements. Loopback OAM cells may be placed within operational VCs and/or VPs.

The ATM layer UNI traffic and congestion control mechanisms are a subset of the B-ISDN traffic and congestion control mechanisms. Traffic control is a means of avoiding network congestion on an uncongested network. Congestion control is a means of minimizing existing congestion problems caused by network failures and unpredicted traffic bursts. Both traffic and congestion controls are used to meet QOS agreements established during connection setups.

ATM layer traffic control, and congestion control within UNI 3.1 define the following functions:

1. Network Resource Management (NRM). - Defines the use of VPCs used at the UNI to simplify CAC, and traffic segregation and priority.
2. Connection Admission Control (CAC). - Defines admission control mechanisms for new VPCs or VCCs requested. These control mechanisms are based on allowed, and available network resource.
3. Usage Parameter Control (UPC). - Required only at a Public UNI to protect a network from malicious or unintentional misbehavior which can affect the QOS of other already established connections. UPC does this by monitoring the validity of Virtual Path Connections (VPC) or Virtual Channel Connections (VCC) identifiers and making sure that all associated VPC or VCC traffic entering a network does not violate pre-established traffic contracts. This could be referred to as "Policing." An UPC can perform cell passing (for conforming cells), cell tagging (optionally implemented on cells with CLP=0 only by making the CLP = 1), or cell discarding (for non-conforming cells). Excessive policing can degregate network performance[7].
4. Selective Cell Discarding (SCD). - Allows cells with CLP = 1 to be discarded in the event of congestion.
5. Traffic shaping. - Previously discussed.
6. Explicit Forward Congestion Indication (EFCI). - Used by an application above the ATM layer to control its own traffic parameters in the event of network congestion. This feature allows

network elements to set explicit forward congestion indication bits in cell headers so that applications above the ATM layer could self regulate their traffic.

C. RFC 1695 (ATOM MIB) DEFINITIONS OF MANAGED OBJECTS FOR ATM MANAGEMENT

The AToM MIB defines a set of objects to manage ATM interfaces, devices, networks, and services. Like the LECMv2, and LESMv1, it is based on SNMPv2 protocol. Specifically it manages the ATM Cell layer, and AAL5 CPCS. These are denoted by the ifType values "atm(37)," and "aal5(49)," respectively within MIB-II ifTable row entries.

The interfaces group of MIB-II specifies protocols by row numbers within an interface table "ifTable." Each row is identified by an "ifIndex" value, and has a number of associated row entries, one of which, is an interface Type specified as "ifType." IfType values examples include "atm(37)," "sonet(39)," "aal5(49)," and "aflane8023(59)." The aflane8023(59) identifies an interface as belonging to an 802.3 Emulated LAN layer. The International Assigned Numbers Authority (IANA) periodically publishes the ifType values. The hierarchical relationships between row entries in the ifTable are found in an interface stack table denoted by "ifStackTable." Each row in ifStackTable points to the upper and subordinate layer associated row entries, for each row entry within the ifTable.

The ATM managed objects within the AToM MIB are arranged within groups. A listing and general description of each group function is as follows:

1. ATM interface configuration group. – Provides ATM cell layer configuration associated with ifTable references.
2. ATM interface DS3 PLCP group. – Provides performance statistics of DS3 type interfaces.
3. ATM interface TC sublayer group. – Provides performance statistics of an interface Transmission Convergence (TC) sublayer.
4. ATM interface virtual link (VPL/VCL) configuration group. – Provides the creation, deletion, or modification of links on ATM hosts, or ATM switch network devices.
5. ATM VP/VC cross-connect group. – Provides the creation, deletion, or modification of cross-connects on ATM switch network devices.
6. AAL5 connection performance statistics group. – Provides performance statistics related to the AAL5 CPCS.

D. SDUS AND PDUS

When a lower layer receives information from an upper layer, that information is called a (lower layer name) - Service Data Unit (SDU) to that lower layer. The upper layer calls the information it delivers to a lower layer a (upper layer name) - Protocol Data Unit (PDU). Note, that both of these terms (upper layer name) – PDU, and (lower layer name) - SDU refer to the same information, but each layer names the information it receives, or delivers, relative to itself.

E. USER PLANE - LANE, AND ATM

An example showing the layers involved with the transfer of user information between EES-1, and EES-4 across NetX (Figure 1) are depicted in Figure 15. The interconnections between AED-1, and AED-2, through AS 1-3 are established by LANE, UNI, and PNNI protocols before EES data can be sent. Figure 15 does not show any control type layers, nor ATM services involved with the setup, nor control of these ATM connections. The ATM backbone switches (not including edge devices) use only the ATM, and Physical layers (User Plane) for user data transport.

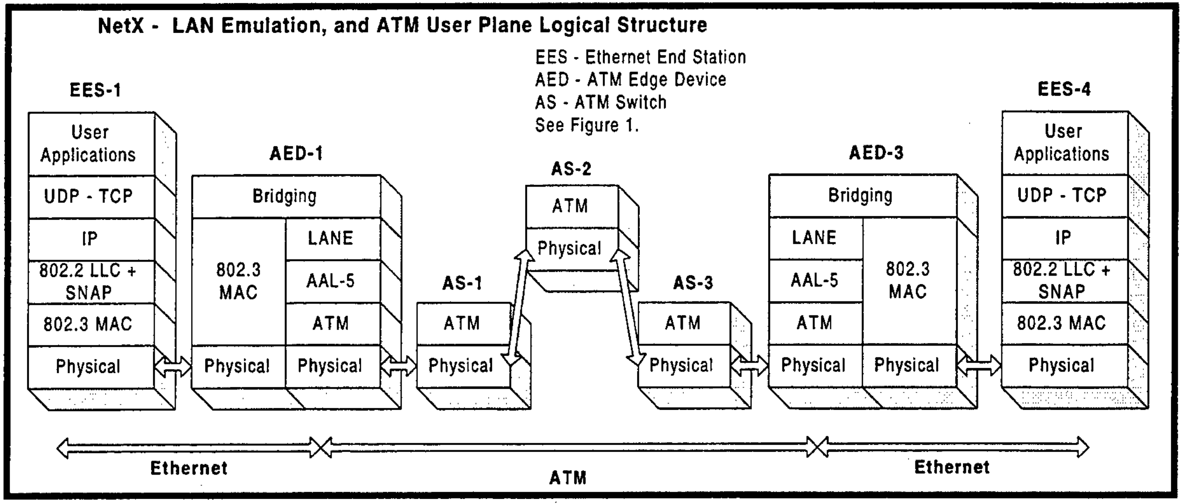


Figure 15. User Plane – Protocol Layers (EES, AED, and AS)

The Bridge, in Figure 15, is a representation of a simple passing mechanism an edge device has for passing Ethernet frames to and from the LANE layer and 802.3 MAC Layer.

1. LANE and the ATM User Plane

An exploded view of the ATM layers associated with the ATM side of an edge device is shown in Figure 16. The left-hand column in the figure is a high level view of each layer with associated details to the right. Note that LANE layer (as described in Figure 15) is referred to as a LAN Emulation Client (LEC) in Figure 16. At the bottom of Figure 16 is the layout of a SONET frame sent between ATM interfaces operating at 155 Mbps.

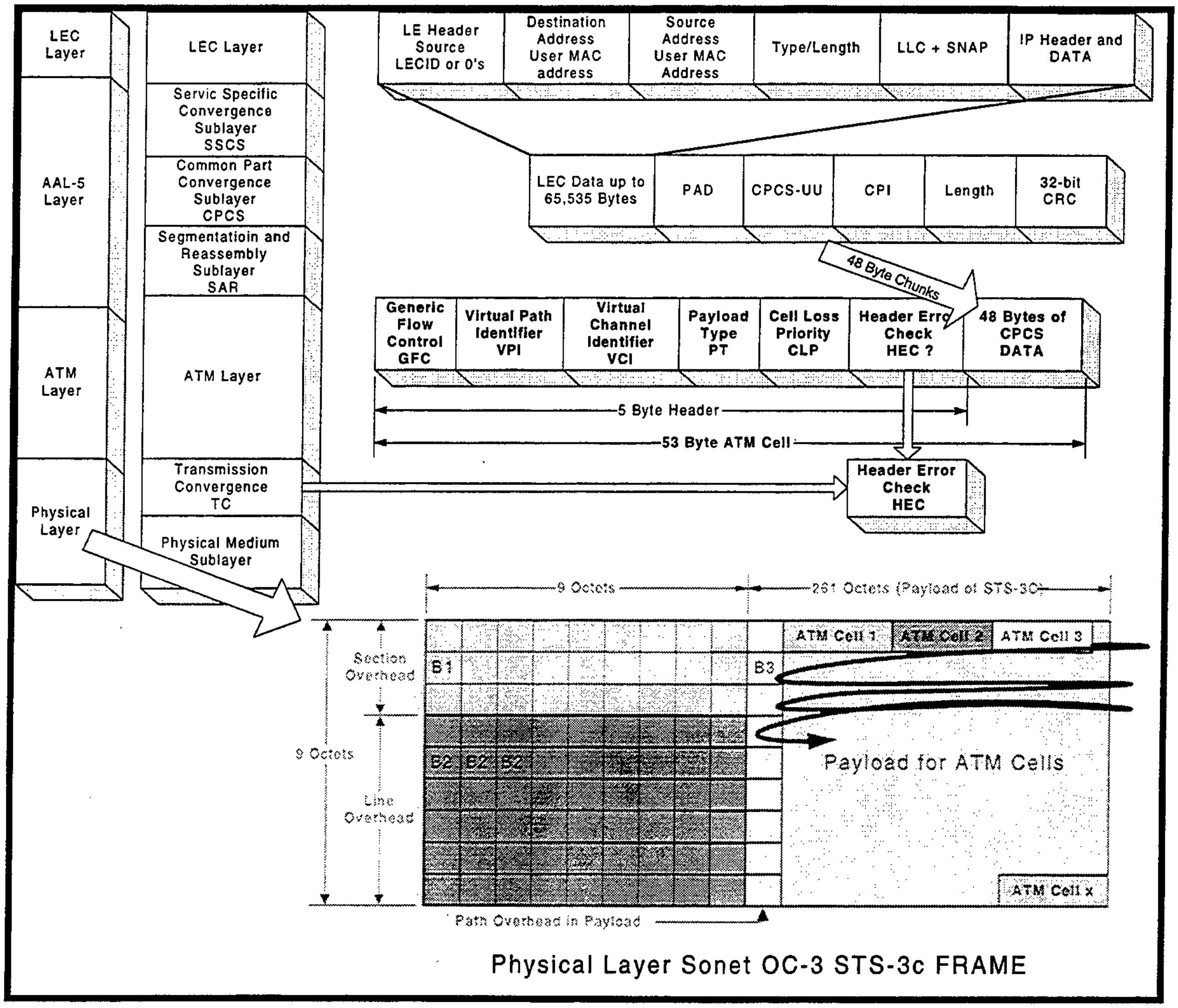


Figure 16. User Plane – Layer Details

The AAL Layer adapts the user traffic from upper layers to the cell-based network. Because different types of traffic have different transfer requirements, in terms of delay, accuracy, timing, etc., this layer is broken in to several types, each associated with an AAL (AAL 1, 2, 3-4, and 5). AAL Type 1 (AAL1) is used for circuit emulation, such as DS1/E1, etc. AAL Types 2, and 3/4 offer other types of services (such as Switched Multimegabit Data Service (SMDS) to ATM mapping, or voice over ATM services) that will not be described in this document. AAL3/4, and 5 have a Convergence Sublayer composed of a Service Specific Convergence Sublayer (SSCS), and a Common Part Convergence Sublayer (CPCS). SSCS is used to provide additional services, such as "assured mode" services to signaling protocols, or others to support applications like Frame Relay, or SMDS. The CPCS is used primarily to provide an error detection capability. AAL5 supports variable bit rate (VBR) traffic, both connection-oriented, and connectionless. AAL5 is used for LAN data transport, and signaling. A LANE LEC interfaces directly with AAL5's CPCS. LANE does not use the SSCS for user data transport, only control signaling.

2. LANE (LEC) Layer

As can be seen in Figure 16, a LANE LEC takes an Ethernet frame (max 1514 bytes without its 4 byte CRC), adds a 2 byte LECID field and delivers it (called the LEC-PDU) to the AAL5 CPCS. If more than one frame were to be sent to the same end destination by a LEC, the frames could be encapsulated in the same CPCS data payload (which can range in size from 1 to 65,535 bytes).

3. CPCS Layer

The CPCS adds five fields to the LEC-PDU: (1) PAD, (2) Common Part Convergence Sublayer – User-to-User (CPCS-UU), (3) Common Part Indicator (CPI), (4) Length, and (5) a 32-bit Cyclic Redundancy Check (CRC) field. The PAD field is used to add from 0 to 47 additional bytes to the CPCS-PDU so that the entire length of the CPCS-PDU (which is made up of the CPCS-SDU (data payload), PAD field, CPCS-UU, CPI, Length, and CRC fields) is an integer number of 48 bytes (for SAR). The CPCS-UU field is used to identify the user payloads between AAL users transparently. The CPI is used to align the CPCS trailer to a 64-bit boundary. The Length field is used to identify the length of the payload (not including the length of the PAD field) so that the receiver can detect loss or gain. The Length field, if coded as zero, requests that a partially filled CPCS-SDU be discarded by the receiver (some applications may use this when problems occur). The 32-bit CRC field is the same as that used by 802.3, and is filled with a CRC-32 result, which is processed over the CPCS PDU (excluding the CRC field itself). Once the CPCS-PDU is created, it is handed to the Segmentation And Reassembly (SAR) sublayer.

4. SAR Layer

The AAL5 (SAR) sublayer simply divides the CPCS-PDU into 48-byte chunks, calculates information for the appropriate ATM Payload Type field, and sends this to the ATM layer. The 48-byte payload, together with the Payload Type (PT) information that is passed to the ATM layer is called the AAL-Protocol Data Unit (AAL-PDU). Note that the AAL5 SAR also provides for the passing of congestion and loss priority information between the layers above and below it.

F. CONTROL PLANE – LANE, AND ATM

As previously discussed the C-Plane provides call establishment, release, and other connection control functions necessary for providing switched services. This plane has AAL procedures, higher layer signaling protocols, and shares the Physical, and ATM Layers with the U-Plane. The circuits associated with the control plane are called Switched Virtual Circuits (SVCs). SVCs are set up on-request in the ATM network, so that an ATM communication path is established between the requestor and the destination associated ATM devices. A Permanent Virtual Circuit (PVC) is an ATM communications path that is statically set-up, or torn-down by a network administrator (in each ATM device, from an ATM source, to an ATM destination), and does not require the C-Plane.

A SVC, or PVC, can either be a bi-directional point-to-point or a uni-directional point-to-multipoint virtual connection, at either the VP and/or VC level. For ATM VP users, the user is responsible for allocating VCs within the VP Connection (VPC) provided by the network. For ATM VC users, the network is responsible for allocating the VCs within the VC Connection (VCC) for the user. The VPC, or VCC associated with SVCs are established and released via signaling.

C-Plane procedures required to establish a connection are dependent upon a number of signaling protocols including those in the UNI standard, and those in the PNNI standard.

EESs on NETX (Figure 1) are not directly involved, nor have access to ATM signaling protocols. Figure 17 shows a high level view of where UNI, and PNNI protocols are used.

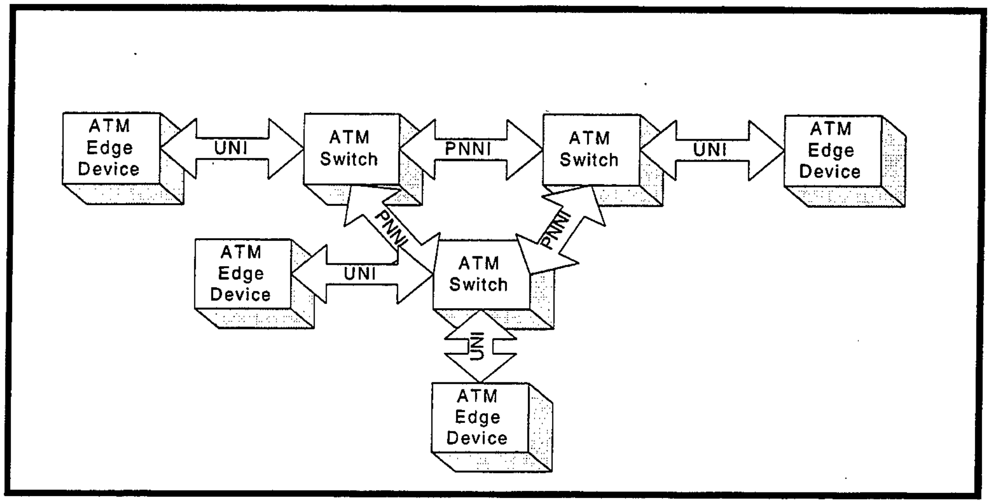


Figure 17. Control Plane – UNI and PNNI Placement in Network

Figure 18 shows the layers involved with the UNI, and PNNI standards. Each Edge Device, and physically connected ATM switch use the UNI signaling protocol to connect, maintain, and clear ATM circuits. Between ATM switches, the PNNI signaling (NNI), as specified in the PNNI standard, is used. An exploded view of the Signaling AAL (SAAL) is shown at the far left of the figure. The UNI 3.1 standard is described first, then the PNNI standard.

1. UNI 3.1 Signaling

The UNI 3.1 signaling standard is based on the ITU-T Recommendation Q.2931 signaling procedures to establish, release, and monitor ATM connections. High level descriptions of layer functions associated with UNI (as shown in Figure 18) are as follows:

1. Q.2931 is a ITU-T signaling protocol that is used to setup, tear-down, and monitor UNI VCCs.
2. SSCF is specified in ITU-T Q.2130, and maps the services of the SSCOP to the needs of the Q.2931 signaling procedures.
3. SSCOP provides for the recovery of lost or corrupted signaling information, supports variable length traffic, and is specified in ITU-T Q.2110.

4. AAL5 CPCS is specified in ITU-T I.363 and provides information integrity and other services to higher layers.
5. AAL5 SAR also specified in ITU-T I.363 provides 48 Byte payload segmentation and reassembly as well as payload type information. The ATM and Physical layer functions were previously described.

Details about Q.2931 (UNI 3.1) are discussed

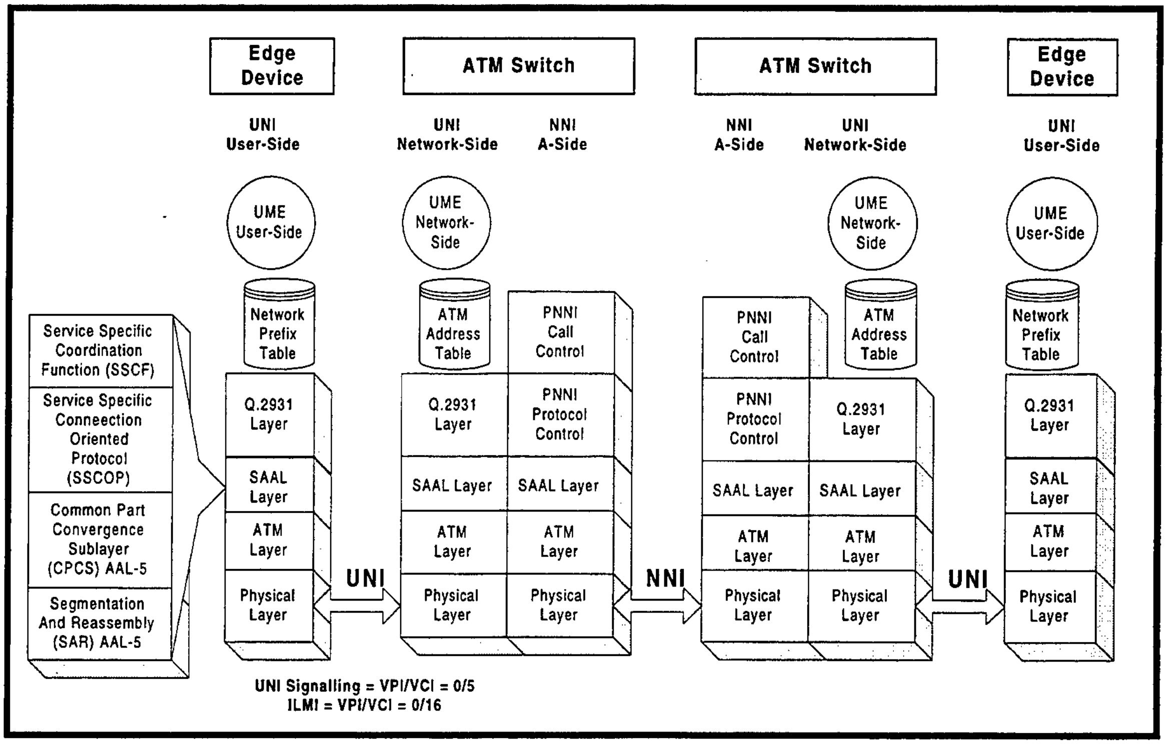


Figure 18. Control Plane – AED and AS Layers and Management Information

a. SSCF, SSCOP, and CPCS

The UNI SAAL layer, shown in Figure 18, provides reliable transport of UNI (Q.2931) messages between peer Q.2931 entities. The structure of the SAAL is defined in ITU-T Q.2100. The SAAL, which is a peer-to-peer protocol is divided into a Service Specific Coordination Function (SSCF), the Service Specific Connection Oriented Protocol (SSCOP), the Common Part Convergence Protocol Sublayer of AAL-5 (CPCS), and the AAL-5 Segmentation and Reassembly (SAR) sublayer. The ATM and Physical Layer functions are the same as those of the U-Plane.

With respect to the SAAL layer, a service application such as Q.2931 is considered a user of SAAL services. The SSCOP sublayer provides the main functionality of the SAAL with SSCFs providing application specific mapping to the SSCOP services. A SSCF is service application specific and in Figure 18 provides the mapping of services between Q.2931, and the SSCOP sublayer. Depending upon the service application the SSCF could be functionally null. The SSCOP can also run over different AAL Common Part Protocols (specified in ITU-T I.363).

An application can request unassured data transfer, or assured data transfer; these are referred to as "unassured mode," or "assured mode" connections by SSCOP respectively. With unassured mode, data sent from an application is simply sent to its peer via SSCOP without the expectation of the receipt of this transmission by its peer to be acknowledged. A peer SSCOP state machine can be in any state, when unassured mode transmissions are transmitted or received. PNNI signaling uses assured mode for data transfer. To see an example of a PNNI SETUP message encapsulation by the ATM associated layers look at Figure 32.

Q.2931 (UNI and PNNI are based on this) requires assured mode signaling. With assured mode, SSCOP peers first set up a connection with one another, and monitor this connection status as long as it is needed. An SSCOP state machine defines how it must act, or react, to message requests or connection conditions. Any message received in an incorrect state, results in error recovery, via ER PDUs, with acknowledgement from peer, or a Resynchronization via RS-PDU, also with acknowledgement from peer. All SSCOP messages are numbered sequentially. POLL messages are sent periodically requesting the status of the messages, buffers, and the state of the receiving SSCOP. Timers are also kept along with state variable to keep track of which messages and polls were sent but not acknowledged. The receiver has timers and state variables to track the status of what messages and polls were not received. The receiver responds to a missing message with an unsolicited status message (USTAT PDU) telling the source that it did not receive a specific message. The receiver responds to POLL messages with aggregate status (STAT PDU) information on what was or was not received, along with its own buffer status to regulate transmission rate.

Each message PDU sent has a PDU type field, a PAD field, a Reserved field, along with other information fields specific to the PDU type. The PAD field makes sure that the PDU message is a multiple of 4 octets long. The PDU type field identifies the PDU as one of 14 different types. The Reserved field primarily is used to 32 bit align the fields within the PDU. If there is an unrecognized PDU type code, an incorrect length given PDU type, or if the fields are not 32 bit aligned within the PDU, the PDU is classified as "invalid" and discarded. Invalid PDUs are discarded without notification to the sender.

If a receiver receives a PDU with a sequence number that indicates a lost PDU, it sends an unsolicited status USTAT- PDU message to the receiver telling it which PDU was not received. The source also sends periodic messages (POLL–PDUs) to the receiver, with transmitter state information and requests for receiver information. The receiver responds with it's state information, credit information (I have buffer space for you to send x more messages), as well as information regarding missing PDUs. The transmitter retransmits messages (PDUs) that are lost.

The SSCOP uses the CPCS (with the corrupted data delivery option specified in ANNEX E of ITU-T I.365.5) and SAR for information transfer and data error detection.

The CPCS layer PDU is shown in Figure 16 and its operation addressed. A CPCS option called the "corrupted data delivery" used by UNI and PNNI provides for the delivery of a "Reception Status" (RS) parameter with every delivered CPCS-SDU (when the CPCS delivers a CPCS-SDU to the SSCOP). This RS parameter is a flag which lists the types of errors detected or an indication of no error detected. Specified flags are as follows: (1) "OK," (2) illegal CRC, (3) illegal CPI, (4) Length field in CPCS-PDU = 0, (5) illegal length of PAD field, (6) Length field exceeds the Maximum SDU length, (7) Length field exceeds corrupted length parameter, or (8) reassembly timer expiration prior to the completion of the CPCS-SDU assembly. The SSCOP is responsible for determining the impact to the CPCS-SDU if the error can not be accepted.

b. UNI Signaling Message Format

The UNI signaling message format with associated field values, as per the UNI 3.1 specification, is shown in Figure 19. The structure of the UNI signaling message is the same as that for PNNI, and Q.2931 (see Figure 33) with Protocol Discriminator Field identifying each. Details about the use and operation of these fields are found in Section 2.b(2) of this Chapter. UNI signaling message encapsulation is the same as that for PNNI signaling messages, shown in Figure 32 (with the UNI signaling layer, and messages replacing the PNNI signaling layer, and messages above the SSCF). PNNI does add additional message types, and Informational Element field values, among other things to facilitate routing.

Each Informational Element (IE) in a UNI or PNNI signaling message has its own field format which begins with an IE Identifier field. The IE Identifier field values are also shown in Figure 19. Please note that IE "0101 1111" identifies an IE as a Broadband Low-Layer Information (BLLI) IE. The BLLI IE is used by LANE to identify the type of VCC that is in reference. This document references the BLLI (Informational Element

c. ILMI

A special assured mode signaling (a feature provided by the SAAL) VCC, with VPI/VCI equal to 0/5, is standardized. To assist Q.2931 signaling, a UNI Management Entity (UME) is created at each interface, and an Interim Local Management Interface (ILMI) communication protocol is used. ILMI related functions occur over another special standardized VCC, VPI/VCI equal to 0/16. The initial UNI and ILMI VCCs, 0/5, and 0/16 respectively (defined in ITU-T Q.2931), are setup automatically and considered permanent when the device is configured to use the UNI standard.

d. UNI ATM Address Registration Procedure

Each point-to-point UNI connection has a user side and a network side UME. The UME is a management program, which uses the ILMI communication protocol (also specified in the UNI 3.1 standard) to gather status and configuration information to assist Q.2931 in setting up and diagnosing a circuit. Although it is not shown in Figure 18, ILMI uses a Simple Network Management Protocol (SNMP), and defines a Management Information Base (MIB) database at each interface. SNMP is a protocol to use and manipulate information contained in MIB databases. As previously mentioned, a MIB database is created at each interface; a Network Prefix MIB Group Table is created at each user-side UNI, and an ATM Address MIB Group Table is created at each network-side UNI interface. For the purposes of address registration, the MIB icons in Figure 18 only show address registration related tables.

UNI 3.1 Message Format

Protocol Discriminator Manditory (M): 1 Byte (B)	Call Reference Length, Flag, and Value M: 4 B	Message Type M: 2 B	Message Length M: 2 B	Variable length Informational Elements

Message Type Octet 1			
Value	Meaning	Value	Function
0000 0010	CALL PROCEEDING	0111 1101	STATUS
0000 0111	CONNECT	0111 0101	STATUS ENQUIRY
0000 1111	CONNECT ACKNOWLEDGE	1000 0000	ADD PARTY
0000 0101	SETUP	1000 0001	ADD PARTY ACKNOWLEDGE
0100 1101	RELEASE	1000 0010	ADD PARTY REJECT
0101 1010	RELEASE COMPLETE	1000 0011	DROP PARTY
0100 0110	RESTART	1000 0100	DROP PARTY ACKNOWLEDGE
0100 1110	RESTART ACKNOWLEDGE		

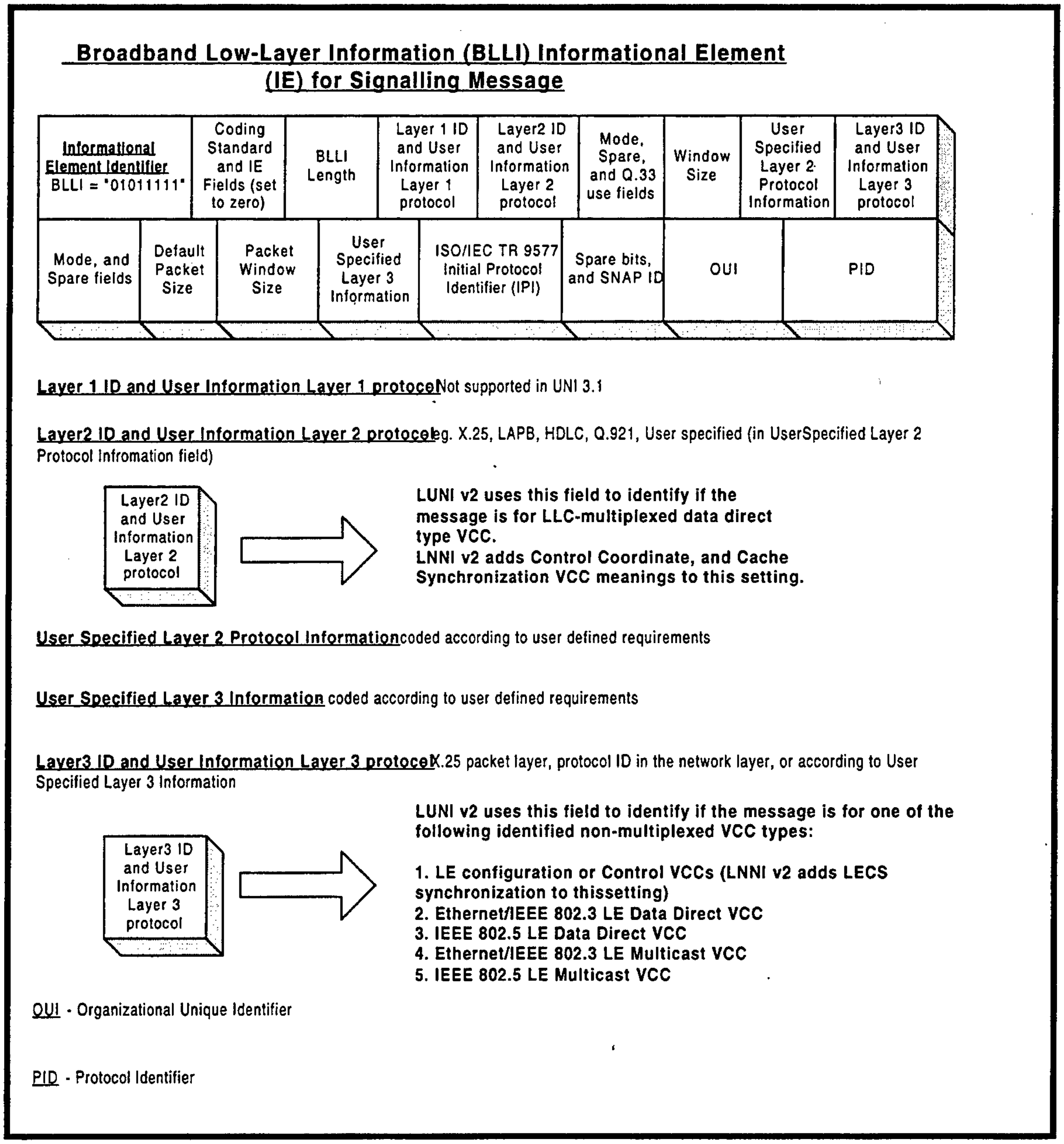
Message Type Octet 2 has a Flag bit and a two bit Action Indicator	
Flag bit	Meaning
"0"	Message Instruction Field not significant
"1"	Follow Explicit Instructions in Message Instruction Field
Action Indicator	
"00"	Clear Call
"01"	Discard and ignore

Message Length - The total number of Bytes for all the Variable length Informational Elements

Each Informational Element included within the message Variable length Informational Element field is identified by a 1 byte Informational Element Identifier field within each Informational Element. Field values and meanings are shown below:

Informational Element Identifier Field			
Value	Informational Element	Value	Informational Element
0000 1000	Cause	0110 0000	Broadband locking shift
0000 0100	Call state	0110 0001	Broadband non-locking shift
0101 0100	Endpoint reference	0110 0010	Broadband sending complete
0101 0101	Endpoint state	0110 0011	Broadband repeat indicator
0101 1000	ATM adaptation layer parameters	0110 1100	Calling party number
0101 1001	ATM traffic descriptor	0110 1101	Calling party subaddress
0101 1010	Connection identifier	0111 0000	Called party number
0101 1100	Quality of service parameter	0111 0001	Called party subaddress
0101 1101	Broadband high layer information	0111 1000	Transit network selection
0101 1110	Broadband bearer capability	0111 1001	Restart indicator
0101 1111	Broadband Low-Layer Information (BLLI)		

Figure 19. The UNI 3.1 Signaling Message Format



Broadband Low-Layer Information (BLLI) Informational Element (IE) for Signalling Message

Informational Element Identifier BLLI = "01011111"	Coding Standard and IE Fields (set to zero)	BLLI Length	Layer 1 ID and User Information Layer 1 protocol	Layer2 ID and User Information Layer 2 protocol	Mode, Spare, and Q.33 use fields	Window Size	User Specified Layer 2 Protocol Information	Layer3 ID and User Information Layer 3 protocol

Mode, and Spare fields	Default Packet Size	Packet Window Size	User Specified Layer 3 Information	ISO/IEC TR 9577 Initial Protocol Identifier (IPI)	Spare bits, and SNAP ID	OUI	PID

Layer 1 ID and User Information Layer 1 protoceNot supported in UNI 3.1

Layer2 ID and User Information Layer 2 protoceleg. X.25, LAPB, HDLC, Q.921, User specified (in UserSpecified Layer 2 Protocol Infromation field)

Layer2 ID and User Information Layer 2 protocol

LUNI v2 uses this field to identify if the message is for LLC-multiplexed data direct type VCC.
LNNI v2 adds Control Coordinate, and Cache Synchronization VCC meanings to this setting.

User Specified Layer 2 Protocol Informationcoded according to user defined requirements

User Specified Layer 3 Information coded according to user defined requirements

Layer3 ID and User Information Layer 3 protoceX.25 packet layer, protocol ID in the network layer, or according to User Specified Layer 3 Information

Layer3 ID and User Information Layer 3 protocol

LUNI v2 uses this field to identify if the message is for one of the following identified non-multiplexed VCC types:

1. **LE configuration or Control VCCs (LNNI v2 adds LECS synchronization to thissetting)**
2. **Ethernet/IEEE 802.3 LE Data Direct VCC**
3. **IEEE 802.5 LE Data Direct VCC**
4. **Ethernet/IEEE 802.3 LE Multicast VCC**
5. **IEEE 802.5 LE Multicast VCC**

OUI - Organizational Unique Identifier

PID - Protocol Identifier

Figure 20. Broadband Low-Level Information (BLLI) Information Element (IE) Format

When a UNI device is first turned on, or restarted, the user-side and network-side UME engage in address registration procedures. An ATM address (see Figure 13) is composed of a prefix and user-part. The prefix is supplied by the network-side of a UNI interface and the user-part is supplied by the user-side of a UNI interface. The prefix is composed of an AFI, ICD, and HODSP. The user-part is composed of an ESI (802.3 MAC address), and a Selector byte. The UNI protocol does not use about the SEL byte, as it exists for user specific functions. Figure 21 shows the UNI ILMI Address Registration procedure.

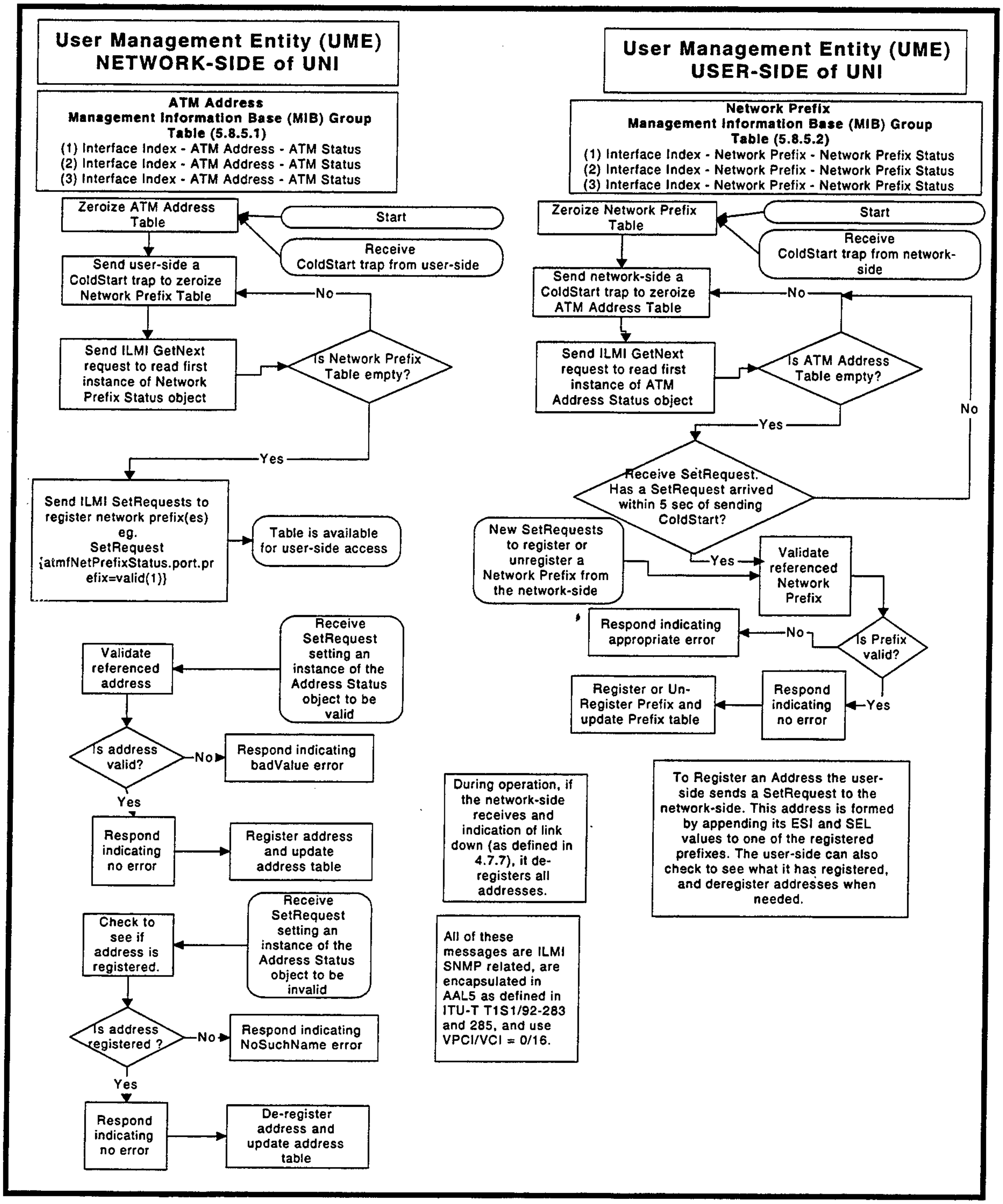

User Management Entity (UME)
NETWORK-SIDE of UNI
ATM Address
Management Information Base (MIB) Group
Table (5.8.5.1)
(1) Interface Index - ATM Address - ATM Status
(2) Interface Index - ATM Address - ATM Status
(3) Interface Index - ATM Address - ATM Status
Zeroize ATM Address Table
Start
Receive ColdStart trap from user-side
Send user-side a ColdStart trap to zeroize Network Prefix Table
No
Send ILMI GetNext request to read first instance of Network Prefix Status object
Is Network Prefix Table empty?
Yes
Send ILMI SetRequests to register network prefix(es) eg. SetRequest {atmfNetPrefixStatus.port.prefix=valid(1)}
Table is available for user-side access
Receive SetRequest setting an instance of the Address Status object to be valid
Validate referenced address
Is address valid?
No
Respond indicating badValue error
Yes
Respond indicating no error
Register address and update address table
Receive SetRequest setting an instance of the Address Status object to be invalid
Check to see if address is registered.
Is address registered ?
No
Respond indicating NoSuchName error
Yes
Respond indicating no error
De-register address and update address table
User Management Entity (UME)
USER-SIDE of UNI
Network Prefix
Management Information Base (MIB) Group
Table (5.8.5.2)
(1) Interface Index - Network Prefix - Network Prefix Status
(2) Interface Index - Network Prefix - Network Prefix Status
(3) Interface Index - Network Prefix - Network Prefix Status
Zeroize Network Prefix Table
Start
Receive ColdStart trap from network-side
Send network-side a ColdStart trap to zeroize ATM Address Table
No
Send ILMI GetNext request to read first instance of ATM Address Status object
Is ATM Address Table empty?
Yes
No
Receive SetRequest. Has a SetRequest arrived within 5 sec of sending ColdStart?
New SetRequests to register or unregister a Network Prefix from the network-side
Yes
Validate referenced Network Prefix
Is Prefix valid?
No
Respond indicating appropriate error
Yes
Respond indicating no error
Register or Un-Register Prefix and update Prefix table
During operation, if the network-side receives and indication of link down (as defined in 4.7.7), it de-registers all addresses.
To Register an Address the user-side sends a SetRequest to the network-side. This address is formed by appending its ESI and SEL values to one of the registered prefixes. The user-side can also check to see what it has registered, and deregister addresses when needed.
All of these messages are ILMI SNMP related, are encapsulated in AAL5 as defined in ITU-T T1S1/92-283 and 285, and use VPCI/VCI = 0/16.

Figure 21. Control Plane - UNI 3.1 ILMI Addressing Process

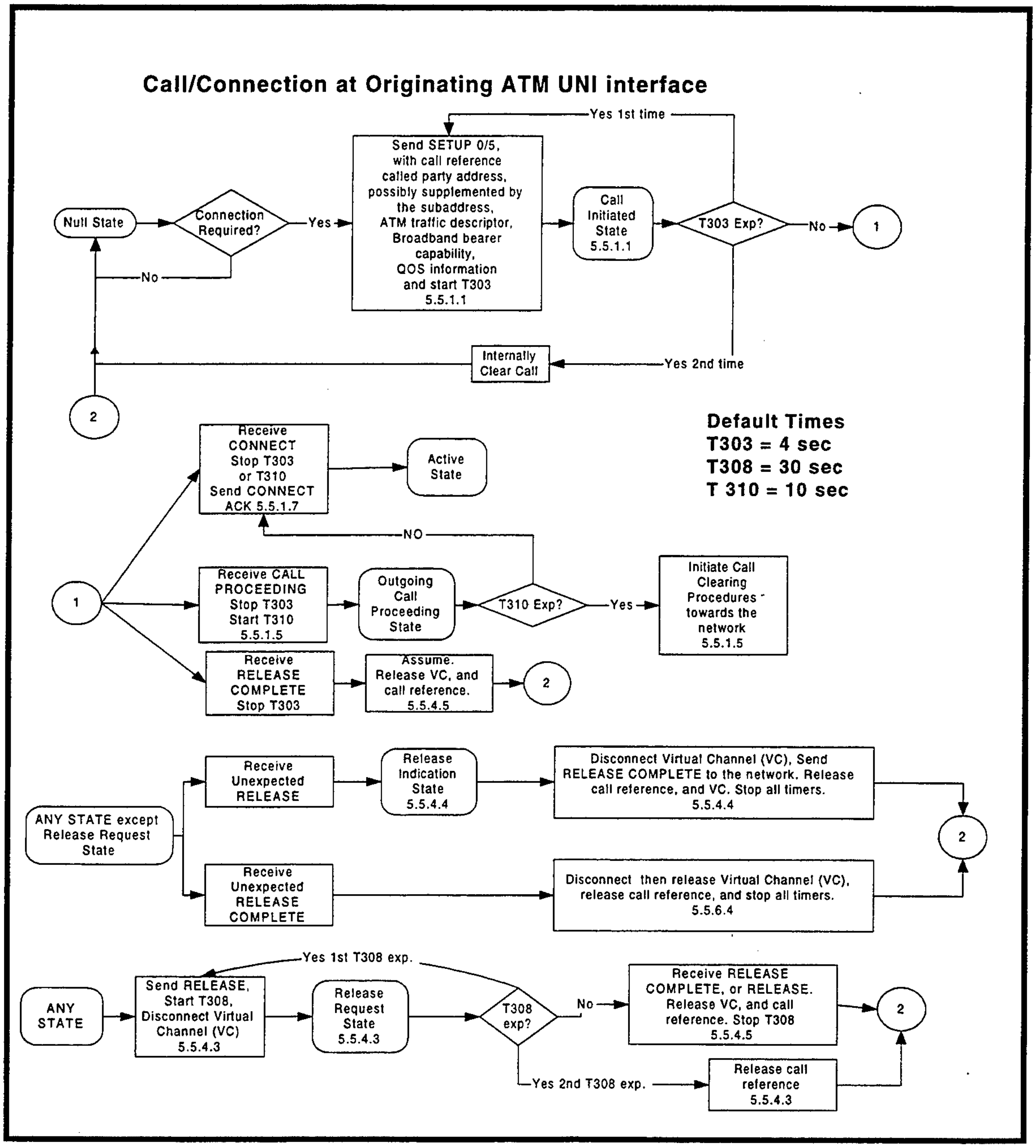

Call/Connection at Originating ATM UNI interface
Yes 1st time
Send SETUP 0/5, with call reference called party address, possibly supplemented by the subaddress, ATM traffic descriptor, Broadband bearer capability, QOS information and start T303 5.5.1.1
Null State
Connection Required?
Yes
No
Call Initiated State 5.5.1.1
T303 Exp?
No
1
Internally Clear Call
Yes 2nd time
2
Default Times
T303 = 4 sec
T308 = 30 sec
T 310 = 10 sec
Receive CONNECT Stop T303 or T310 Send CONNECT ACK 5.5.1.7
Active State
NO
1
Receive CALL PROCEEDING Stop T303 Start T310 5.5.1.5
Outgoing Call Proceeding State
T310 Exp?
Yes
Initiate Call Clearing Procedures towards the network 5.5.1.5
Receive RELEASE COMPLETE Stop T303
Assume. Release VC, and call reference. 5.5.4.5
2
ANY STATE except Release Request State
Receive Unexpected RELEASE
Release Indication State 5.5.4.4
Disconnect Virtual Channel (VC), Send RELEASE COMPLETE to the network. Release call reference, and VC. Stop all timers. 5.5.4.4
2
Receive Unexpected RELEASE COMPLETE
Disconnect then release Virtual Channel (VC), release call reference, and stop all timers. 5.5.6.4
Yes 1st T308 exp.
ANY STATE
Send RELEASE, Start T308, Disconnect Virtual Channel (VC) 5.5.4.3
Release Request State 5.5.4.3
T308 exp?
No
Receive RELEASE COMPLETE, or RELEASE. Release VC, and call reference. Stop T308 5.5.4.5
2
Yes 2nd T308 exp.
Release call reference 5.5.4.3

Figure 22. Control Plane – UNI 3.1 Call/Connection at Originating ATM User Interface

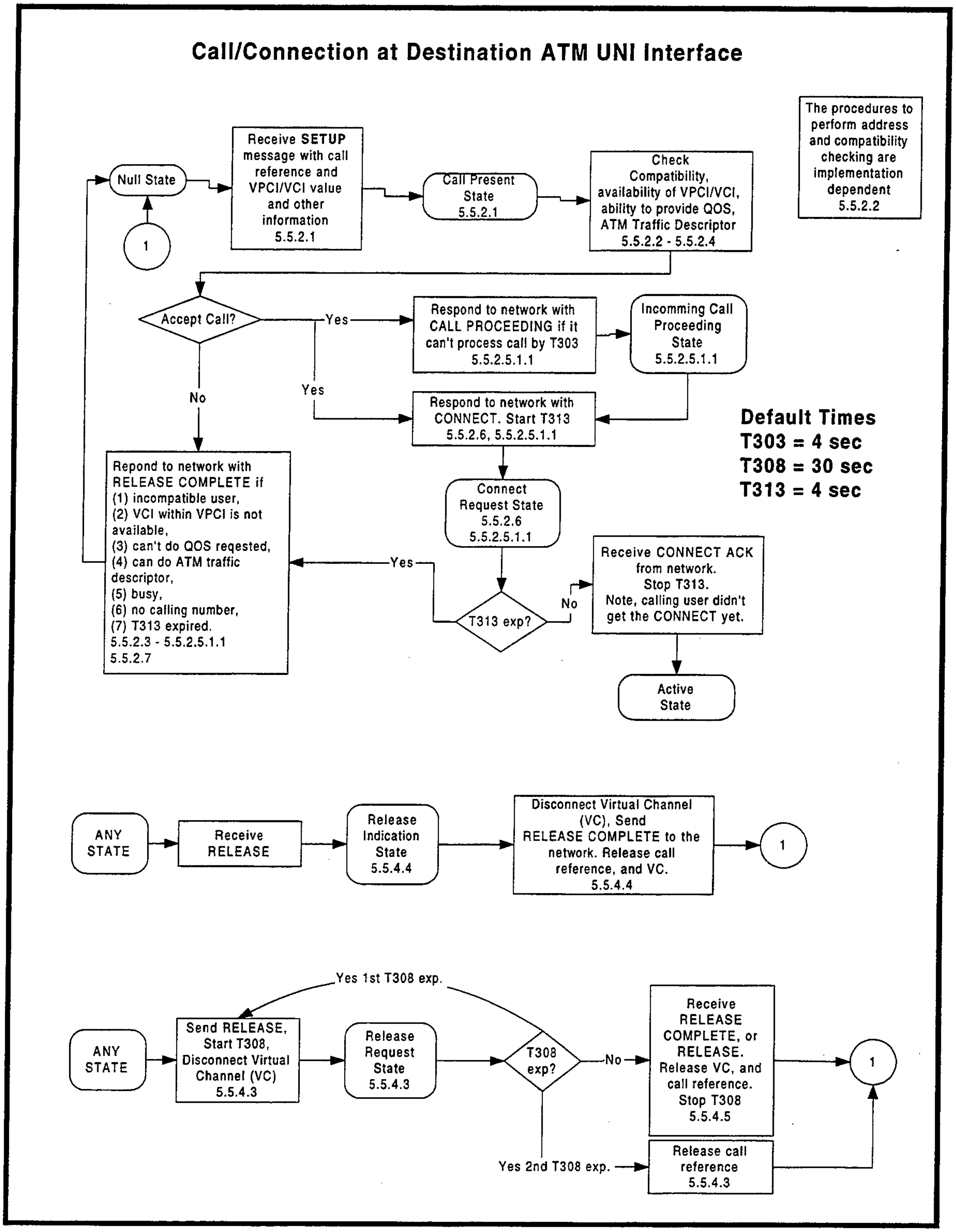

Call/Connection at Destination ATM UNI Interface
Null State
1
Receive SETUP message with call reference and VPCI/VCI value and other information 5.5.2.1
Call Present State 5.5.2.1
Check Compatibility, availability of VPCI/VCI, ability to provide QOS, ATM Traffic Descriptor 5.5.2.2 - 5.5.2.4
The procedures to perform address and compatibility checking are implementation dependent 5.5.2.2
Accept Call?
Yes
Respond to network with CALL PROCEEDING if it can't process call by T303 5.5.2.5.1.1
Incomming Call Proceeding State 5.5.2.5.1.1
No
Yes
Respond to network with CONNECT. Start T313 5.5.2.6, 5.5.2.5.1.1
Default Times
T303 = 4 sec
T308 = 30 sec
T313 = 4 sec
Repond to network with RELEASE COMPLETE if (1) incompatible user, (2) VCI within VPCI is not available, (3) can't do QOS reqested, (4) can do ATM traffic descriptor, (5) busy, (6) no calling number, (7) T313 expired. 5.5.2.3 - 5.5.2.5.1.1 5.5.2.7
Connect Request State 5.5.2.6 5.5.2.5.1.1
Yes
T313 exp?
No
Receive CONNECT ACK from network. Stop T313. Note, calling user didn't get the CONNECT yet.
Active State
ANY STATE
Receive RELEASE
Release Indication State 5.5.4.4
Disconnect Virtual Channel (VC), Send RELEASE COMPLETE to the network. Release call reference, and VC. 5.5.4.4
1
Yes 1st T308 exp.
ANY STATE
Send RELEASE, Start T308, Disconnect Virtual Channel (VC) 5.5.4.3
Release Request State 5.5.4.3
T308 exp?
No
Receive RELEASE COMPLETE, or RELEASE. Release VC, and call reference. Stop T308 5.5.4.5
1
Yes 2nd T308 exp.
Release call reference 5.5.4.3

Figure 23. Control Plane – UNI 3.1 Call/Connection at Destination ATM UNI User Interface

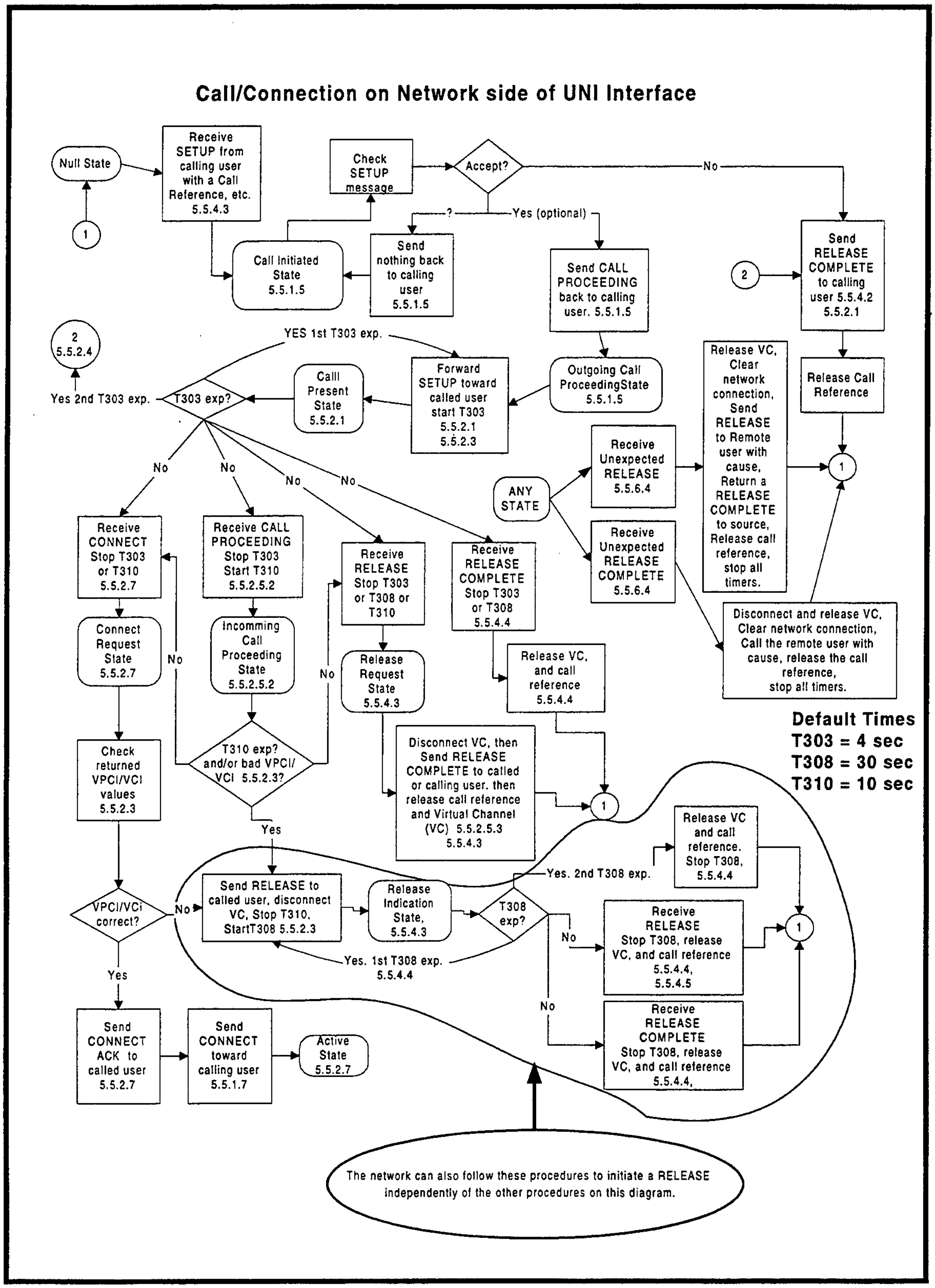

Call/Connection on Network side of UNI Interface
Null State
Receive SETUP from calling user with a Call Reference, etc. 5.5.4.3
1
Check SETUP message
Accept?
No
? Yes (optional)
Call Initiated State 5.5.1.5
Send nothing back to calling user 5.5.1.5
Send CALL PROCEEDING back to calling user. 5.5.1.5
2
Send RELEASE COMPLETE to calling user 5.5.4.2 5.5.2.1
2 5.5.2.4
Yes 2nd T303 exp.
YES 1st T303 exp.
T303 exp?
Calll Present State 5.5.2.1
Forward SETUP toward called user start T303 5.5.2.1 5.5.2.3
Outgoing Call ProceedingState 5.5.1.5
Release VC, Clear network connection, Send RELEASE to Remote user with cause, Return a RELEASE COMPLETE to source, Release call reference, stop all timers.
Release Call Reference
1
No
No
No
No
ANY STATE
Receive Unexpected RELEASE 5.5.6.4
Receive Unexpected RELEASE COMPLETE 5.5.6.4
Receive CONNECT Stop T303 or T310 5.5.2.7
Receive CALL PROCEEDING Stop T303 Start T310 5.5.2.5.2
Receive RELEASE Stop T303 or T308 or T310
Receive RELEASE COMPLETE Stop T303 or T308 5.5.4.4
Disconnect and release VC, Clear network connection, Call the remote user with cause, release the call reference, stop all timers.
Connect Request State 5.5.2.7
No
Incomming Call Proceeding State 5.5.2.5.2
No
Release Request State 5.5.4.3
Release VC, and call reference 5.5.4.4
Default Times
T303 = 4 sec
T308 = 30 sec
T310 = 10 sec
Check returned VPCI/VCI values 5.5.2.3
T310 exp? and/or bad VPCI/ VCI 5.5.2.3?
Disconnect VC, then Send RELEASE COMPLETE to called or calling user. then release call reference and Virtual Channel (VC) 5.5.2.5.3 5.5.4.3
1
Yes
Release VC and call reference. Stop T308, 5.5.4.4
Yes. 2nd T308 exp.
VPCI/VCI correct?
No
Send RELEASE to called user, disconnect VC, Stop T310, StartT308 5.5.2.3
Release Indication State, 5.5.4.3
T308 exp?
No
Receive RELEASE Stop T308, release VC, and call reference 5.5.4.4, 5.5.4.5
1
Yes. 1st T308 exp. 5.5.4.4
Yes
No
Send CONNECT ACK to called user 5.5.2.7
Send CONNECT toward calling user 5.5.1.7
Active State 5.5.2.7
Receive RELEASE COMPLETE Stop T308, release VC, and call reference 5.5.4.4,
The network can also follow these procedures to initiate a RELEASE independently of the other procedures on this diagram.

Figure 24. Control Plane – UNI 3.1 Call/Connection at Network ATM UNI Interface

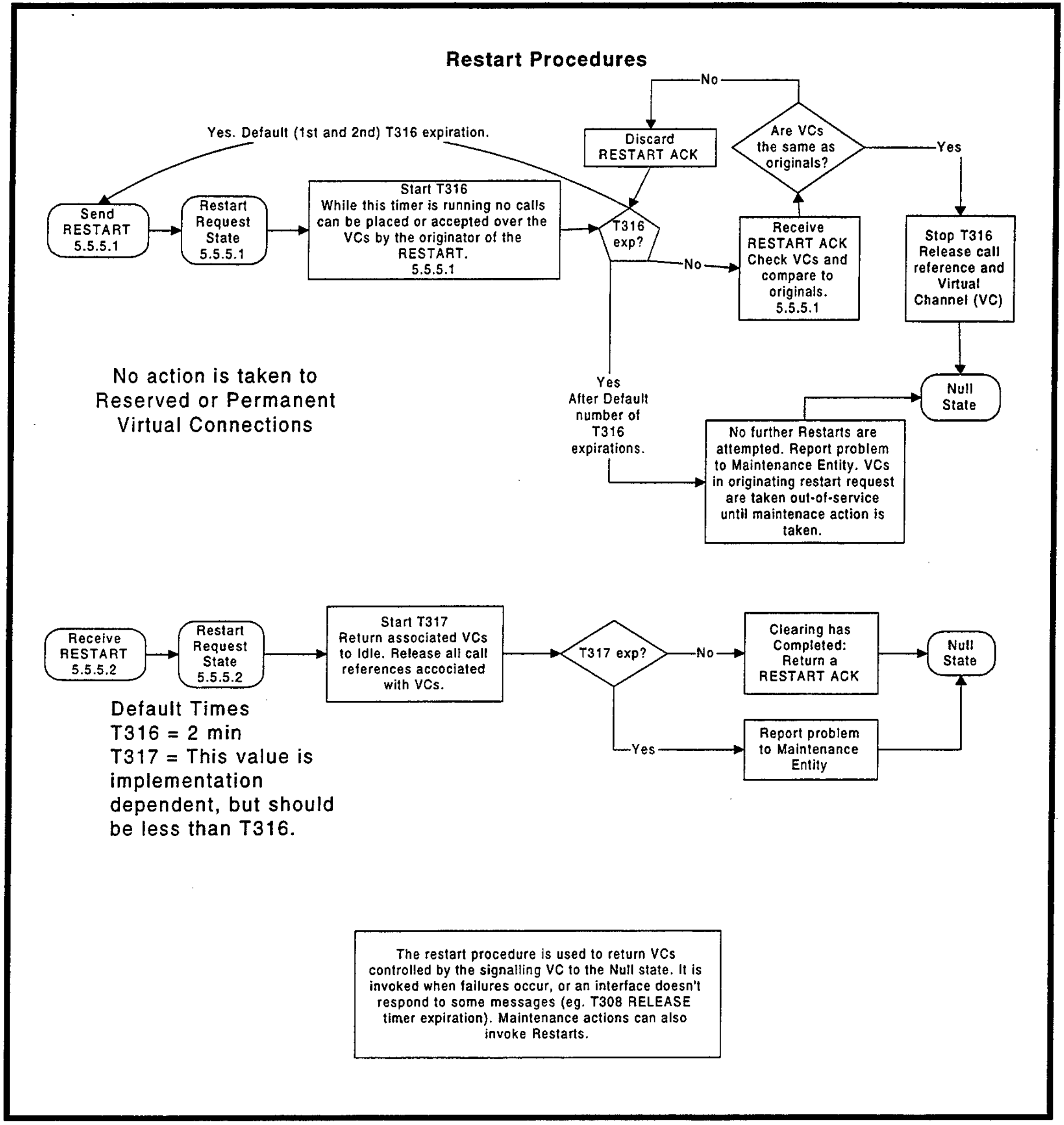

Restart Procedures
No
Yes. Default (1st and 2nd) T316 expiration.
Discard RESTART ACK
Are VCs the same as originals?
Yes
Send RESTART 5.5.5.1
Restart Request State 5.5.5.1
Start T316 While this timer is running no calls can be placed or accepted over the VCs by the originator of the RESTART. 5.5.5.1
T316 exp?
No
Receive RESTART ACK Check VCs and compare to originals. 5.5.5.1
Stop T316 Release call reference and Virtual Channel (VC)
No action is taken to Reserved or Permanent Virtual Connections
Yes After Default number of T316 expirations.
Null State
No further Restarts are attempted. Report problem to Maintenance Entity. VCs in originating restart request are taken out-of-service until maintenace action is taken.
Receive RESTART 5.5.5.2
Restart Request State 5.5.5.2
Start T317 Return associated VCs to Idle. Release all call references accociated with VCs.
T317 exp?
No
Clearing has Completed: Return a RESTART ACK
Null State
Default Times
T316 = 2 min
T317 = This value is implementation dependent, but should be less than T316.
Yes
Report problem to Maintenance Entity
The restart procedure is used to return VCs controlled by the signalling VC to the Null state. It is invoked when failures occur, or an interface doesn't respond to some messages (eg. T308 RELEASE timer expiration). Maintenance actions can also invoke Restarts.

Figure 25. Control Plane – UNI 3.1 Restart Procedures

e. Call/Connection Setup, Release, and Restart

Call/Connection control procedures (point-to-point calls only) are discussed. Point-to-multipoint calls are not covered in this thesis, but are required for LANE v2. As previously stated, a UNI interface has a user-side, and a network-side. There is an ATM originator of a call/connection sometimes referred to as the calling-user, and an ATM destination sometimes referred to as the called-user. The signaling procedures for UNI are complex, and rather than explain them with complicated dialog, I included Figure 22, Figure 23, Figure 24, and Figure 25 so that you can follow how a ATM connection is setup, released, or restarted.

Figure 22 illustrates the call/connection procedures at an originating user-side ATM UNI interface (calling-user). Figure 23 illustrates similar procedures at the destination user-side ATM UNI interface (called-user). Figure 24 illustrates the procedures for any network-side UNI interface. Figure 25 illustrates Restart procedures for any UNI interface.

Figure 8 shows an example of a LANE call setup using UNI. A call or connection (call and connection are synonymous in this section) creation involves "SETUP," "CALL PROCEEDING," "CONNECT," and "CONNECT ACK" messages.

In Figure 8 a source ATM node (LEC A) connects to a destination ATM node (LEC B) by sending a UNI SETUP message to Switch 1. Switch 1, (after determining that it can process the call, forwards a PNNI version of the original SETUP message to Switch 2 and returns a CALL PROCEEDING message to LEC A. Switch 2, after receiving the PNNI SETUP message from Switch 1, and after it determines that it can process the call, sends a PNNI CALL PROCEEDING back to Switch 1. Switch 2 also sends a UNI SETUP message to LEC B. After LEC B determines that it will accept the call, either sends a CONNECT (shown in the figure) or a CALL PROCEEDING (if it needs time to determine if it can process the call – not shown in the figure). When Switch 2 receives the CONNECT message from LEC B, it responds with a CONNECT ACK to LEC B, and forwards the CONNECT message to Switch 1. At this point the signaling layer at LEC B thinks that LEC A has acknowledged the CONNECT message and may begin sending data to it, even though the data may never reach LEC A. This is a problem which LANE fixes with LANE READY_IND, and READY_QUERY messages discussed in Section III.D.9. When Switch 1 receives the CONNECT message from Switch 2 it forwards it to LEC A, which (optionally) replies with a CONNECT_ACK. The call is now setup end to end.

A connection Release (the giving up of VPI/VCI, and call-reference values) involves "RELEASE," and/or "RELEASE COMPLETE" messages shown if Figure 22, Figure 23, and Figure 24. The UNI "STATUS," and "STATUS ENQUIRY" messages are not covered in this document.

It must be restated that the UNI messages noted above, originating from the Q.2931layer use the SAAL assured mode connectivity feature. ILMI messages do not use the Service Specific parts of the SAAL and therefore are not over an assured mode connection.

2. PNNI Routing and Signaling

The PNNI standard defines both routing and signaling functions within a network backbone. It is used between public or private network ATM switches. The PNNI routing specification covers: (1) the discovery of neighbors and link status, (2) synchronization of topology databases, (3) summarization and flooding of topology information, (4) the definition of a routing hierarchy, and the election of "Group Leaders." The PNNI signaling specification, like the UNI signaling protocol, contains the procedures to setup, release, and maintain ATM connections between Private Network-Network Interfaces (between ATM switches).

a. PNNI Routing

PNNI defines peer groups and a routing hierarchy to reduce the amount of routing information that each node must maintain to process connection requests. All nodes within a peer group have identical routing tables that contain detailed information about each physical link and node within that peer group. In addition to this, each peer group has summarized reachability information to connect to any peer group within, and outside a PNNI routing domain.

The topologies of a PNNI hierarchy are manually configured. The plural of topology stated in the previous sentence is not a mistake. A network designer manually configures PNNI peer groups via addressing, levels, and assigns priorities, and capabilities of nodes within each peer group to enable the network to heal itself in the event of a failure. If a component of the network should fail, alternative topologies, and functions of nodes within the hierarchy can occur. An example PNNI network is shown in Figure 26.

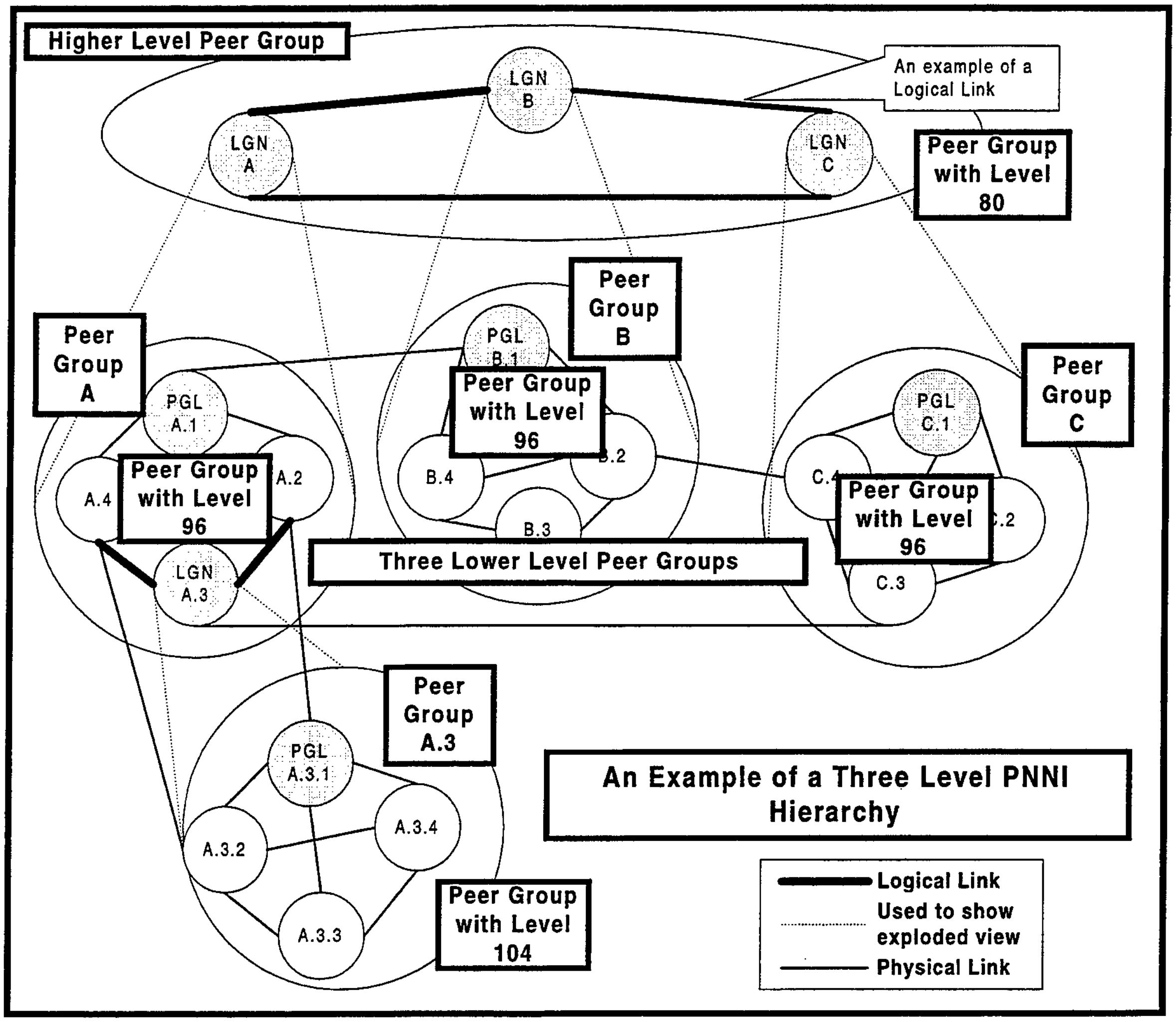


Figure 26. PNNI – Hierarchical Peer Group Structure

(1) Addressing and Hierarchy. The hierarchy of a PNNI network is defined by the commonality of the number of Most Significant Address Bits (MSABs -left justified) in the first 13 Bytes of the 20 Byte ATM addresses assigned to members of a peer group. The number of common MSABs in a peer group is its "level." If 104 MSABs are the same for each node in a peer group then the level is 104 for that peer group. Peer Group A.3 in Figure 26 is a level 104 peer group. A peer group (PG) can have Parent PGs (PPGs) and/or Children PGs (CPGs). A PPG has a lower level number, but is at a higher level than its child PG. An example of a PPG to PG A.3 is PG A. PG A has a level number of 96, which is smaller than its CPG level number. As can be deduced, a CPG has a larger level number than its parent. In Figure 26 the highest level PG has a level (number) of 80, the lowest PG (PG A.3) has a level (number) of 104.

See Figure 27. A PG-Identifier (PGID) is 14 bytes long, and consists of one octet specifying the peer level number (0-104), and 13 Bytes (Address Prefix) of peer group identifier information. A node that uses "native" addressing within a PG will have the same address prefix of every other native addressed member of the peer group. A node that uses "foreign" addressing will have a different address prefix than the natively addressed nodes within its peer group.

Each physical node in a network is called a Lowest Level Node. All nodes in a PNNI hierarchy are called logical nodes, even the Lowest Level Nodes. The reason all nodes are called logical nodes in a PNNI hierarchy is that multiple nodes can be configured on a single physical node; they do not have to be separate physical devices. For example, in Figure 26 the physical ATM switch that is identified as a lowest level node A.3.1, also has node A.3 configured on it. The point is that a single switch can implement multiple logical nodes.

Each PG has a Peer Group Leader (PGL) that is elected via a continuously running election process. A node is manually configured with a "leadership priority number" (LPN). If the LPN is set to zero, it cannot be a PGL. If the LPN is non-zero, then the node with the highest configured LPN is elected, with the higher node id used as selecting a PGL in the event of a tie.

Each PGL has an abstract representative node configured in it to represent that PG in its PPG. This abstract representative node is called a Logical Group Node (LGN). As can be seen in Figure 26, A.3.1 was elected as PGL for PG A.3. Node A.3.1 also has LGN A.3 in its PPG.

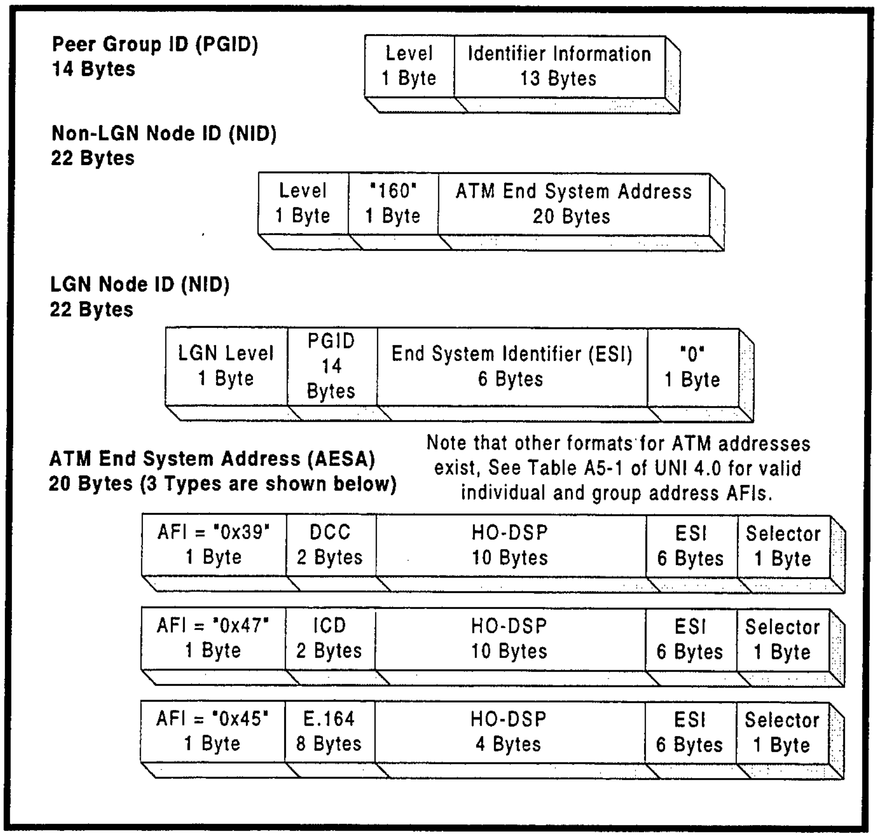
Nodes within a peer group that have physical links to nodes of other peer groups are called border nodes. Physical links to nodes of other peer groups are called an "outside links," and the nodes they connect to are called "outside neighbors." Border nodes identify themselves after an exchange of PNNI Hello packets (discussed in Section (2)). Border nodes do not exchange detailed topology information about their PGs, they only exchange their node-ID, PGID, port-ID, and its nodal hierarchical list so that each may learn the topology of the PNNI network, and possible the reachability of other networks outside the PNNI routing domain. Border nodes tell other members within their PG that they have connections to common ancestor PGs via "Uplinks." Uplinks are not real connections; they are only advertisements about what they can connect too. Higher level ancestors that are in common PGs learn their interconnectivity via Uplinks and "Induced Uplinks (these are higher level uplinks learned from border nodes on lower levels, rather than from border nodes within their PG)." Uplink, and Induced Uplink information are used to establish Routing Control Channels (RCCs) discussed in the next section.

The purpose of a PGL is to summarize detailed routing information about its PG, as well as CPGs, and relay this information to its associated LGN. The LGN shares the summarized information it receives from the PG that it represents, with its peers. When a LGN receives routing related information from its peers, it passes this information down to the PGL, which it represents. The PGL shares the information it receives from its LGN with its peers.

A node within a PG that is not a LGN is a Non-Logical Group Node. The formats of the Logical Group Node Identifiers and Non-Logical Group Node Identifiers differ. See Figure 27. The main difference between the Non-LGN and LGN node-ID formats are that the LGN node-IDs have LGN PG level information, as well as the CPG level information.

If there is only one peer group in a routing domain, the network hierarchy is flat. If a flat network were not connected to any other network, a PGL, and LGN would not be needed for nodes within that PG to communicate. The PGL election process is defined in Section 5.10 in the PNNI V1.0 specification.

ATM End System Addresses (AESAs) can either be individual addresses, or group addresses. The Authority Format Identifier (AFI) field in the first byte of an ATM address is used by PNNI to determine the difference. See UNI 4.0 Table A5-1 for a complete list of valid individual, and group addresses. An individual address specifies a particular end system. A group address specifies more than one end system belonging to a group. PNNI V1.0 is built upon UNI 4.0, which is a superset of UNI 3.1 previously described. UNI 4.0 adds anycast, leaf initiated join, and other operations to PNNI signaling protocol that were not specified in UNI 3.1. Three standard International Code Designator (ICD) values are shown (others exist), which specify the format of the remaining fields in an AESA.



Peer Group ID (PGID)
14 Bytes

Level 1 Byte	Identifier Information 13 Bytes

Non-LGN Node ID (NID)
22 Bytes

Level 1 Byte	"160" 1 Byte	ATM End System Address 20 Bytes

LGN Node ID (NID)
22 Bytes

LGN Level 1 Byte	PGID 14 Bytes	End System Identifier (ESI) 6 Bytes	"0" 1 Byte

ATM End System Address (AESA)
20 Bytes (3 Types are shown below)

Note that other formats for ATM addresses exist, See Table A5-1 of UNI 4.0 for valid individual and group address AFIs.

AFI = "0x39" 1 Byte	DCC 2 Bytes	HO-DSP 10 Bytes	ESI 6 Bytes	Selector 1 Byte

AFI = "0x47" 1 Byte	ICD 2 Bytes	HO-DSP 10 Bytes	ESI 6 Bytes	Selector 1 Byte

AFI = "0x45" 1 Byte	E.164 8 Bytes	HO-DSP 4 Bytes	ESI 6 Bytes	Selector 1 Byte

Figure 27. PNNI and ATM Addressing

(2) Hello Packets, Database Summary Packets, PTSEs, PTSPs, and Topology Tables. After a node is configured, it begins a process of discovery, and further configuration (ex. if elected as a PGL it would create a LGN). Each port that is configured to communicate via PNNI repeatedly transmits "Hello packets" on every configured link. Initially the Hello packets that are sent contain node-ID, port-ID, ATM address information, PGID, and remote node and port-IDs (if known).

After an initial exchange of Hello packets, if a node discovers that it is connected to a node outside its PG, it sends subsequent Hello packets with the following additional information: link aggregation information, its Nodal Hierarchy List, an Uplink Information Attribute (ULIA – this tells the remote node what to advertise about its PG). Hellos between two LGNs send not only the initial Hello packet information, but also LGN Horizontal Link Extension information. Do not get caught up in the terms, as they will be discussed later.

In summary, after an exchange of Hello packets with immediate neighbors, each node knows its neighbors, what peer groups they are in, and other information pertaining to the topology of the PNNI hierarchy. The node then creates one or more PNNI Topology State Elements (PTSEs). PTSEs describe different pieces of the node's environment (links, etc.), and are placed in a node's topology table.

Lowest level nodes in the same peer group, or LGNs that are in the same peer group, then begin a process of synchronizing their topology tables by sending Database Summary (DS) packets to one another. Each DS packet includes summarized information about the PTSEs in each node's topology table. If a node discovers that its topology table is not in sync with that of a its neighbor node's topology table, it requests the required, new, or updated PTSEs from its neighbor. After both nodes have synchronized their databases then the PTSE pertaining to the new link between the nodes is flooded to the remaining nodes in the PG.

The flooding process involves encapsulating all PTSEs in PNNI Topology State Packets (PTSP), making a copy of the PTSP information in the node's Retransmission List, and sending it to each adjacent node associated with that node's PG. The node then waits for each adjacent peer neighbor to acknowledge that it has received every PTSE in a PTSP. Once it receives an acknowledgement for a particular PTSE it removes it from the retransmission list. If no acknowledgement is received the node will retransmit unacknowledged PTSEs and repeat the process.

When a node receives a PTSP it searches its topology table, and retransmission list. The receiving node uses this information to update its topology table, or retransmission list, and sends back an acknowledgement to the sender in the form of a PTSE acknowledgement (PTSE-ACK) packet, or a PTSP (if it needed to update its topology table). If a node had to update a PTSE in its topology table then it continues the flooding by creating a new PTSP, copying the PTSP into a retransmission list, and sending the PTSP to each neighbor. Now this node is originating the PTSP, and waits for acknowledgments. This process continues until all nodes within the PG have identical topology tables. PTSEs in topology tabled expire if not updated within a predetermined amount of time. At the end of this process all nodes within a peer group have synchronized topology tables.

(3) More on Addressing. Earlier I stated that PGL forwards summary address information about its peer group to its LGN. This is not entirely true. See Figure 26. PG A.3's summary address is "A.3"; this provides a means of advertising connectivity to addresses that start with A.3, like A.3.2, etc. An address that is capable of being summarized is called a "Native" address. Note that ATM addresses don't have to be associated with the PGs summary address to belong to that PG. Addresses that don't start with their associated PGs summary address are called "foreign" addresses. A PGL provides summarized native address information, as well as unsummarized foreign address information to the LGN to be broadcast within the LGN PG. An example of an unsummarized foreign ATM address in PG A.3 could be something like M.N.5 (note this is not an ATM address, it's just used to illustrate the concept).

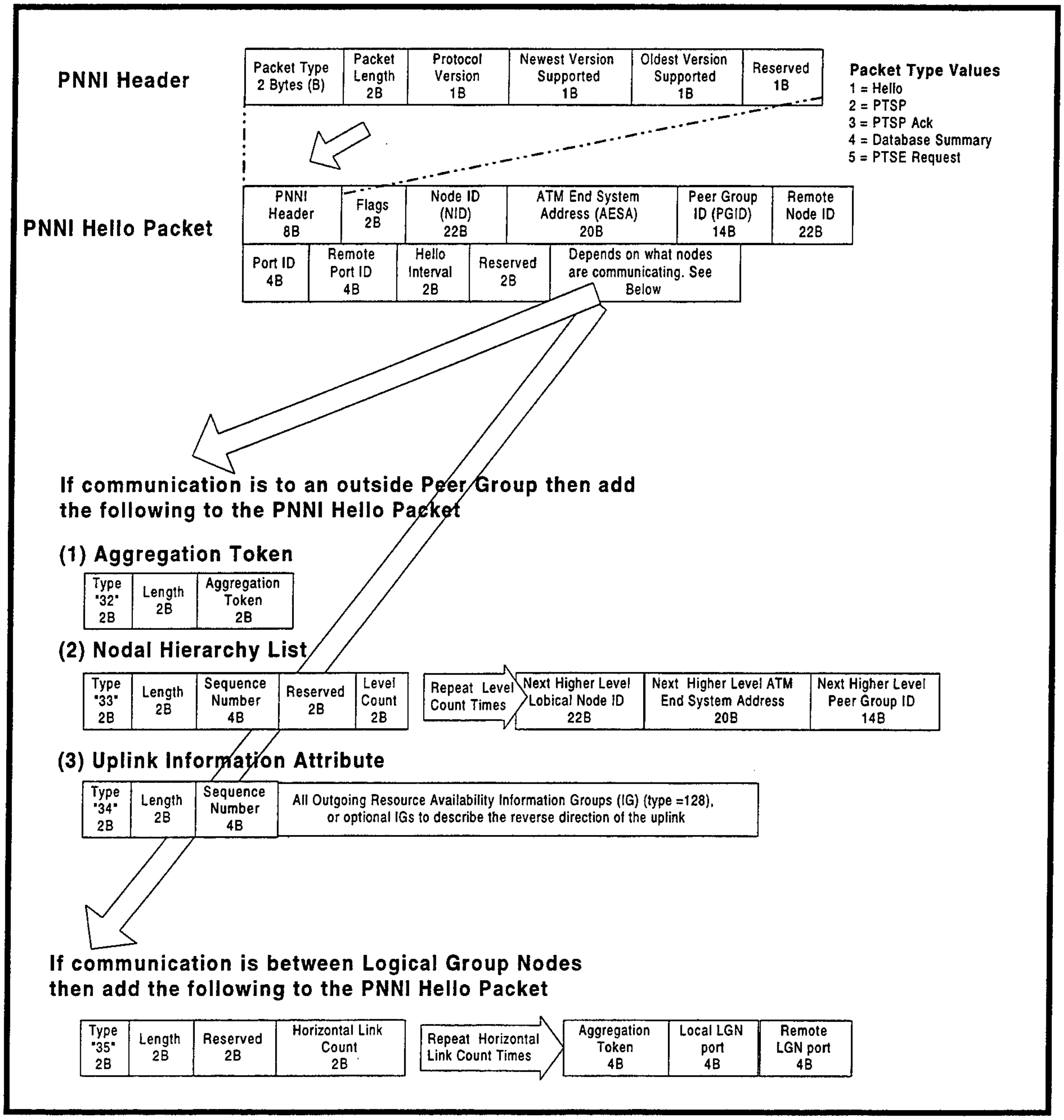
(4) Addressing information that a PG wants advertised originates at a PGL and is forwarded to the LGN. This information is flooded throughout the LGN PG and fed back down from all LGNs in a PG to their children. It is also sent from the LGN PG to its ancestors (higher level parent groups) if not blocked by "scope" (used for individual, and group (anycast) addresses - not discussed in this paper), or "address summary suppression" (a PNNI configuration option to block the advertisement of addresses outside a PG).

PNNI Routing Packet Formats and Routing Control Channels (RCCs). Section (2) described a process of PNNI nodal discovery, and topology construction. This process is carried out using five PNNI packet types: (1) Hello, (2) PNNI Topology State Packet (PTSP), (3) PNNI Topology State Element Acknowledgement (PTSE-ACK), (4) Database Summary (DS), and (5) PTSE Request packet, are shown in Figure 28, Figure 29, and Figure 30. These packets are structured as nested Type-Length-Value (TLV) encodings. TLV's are referred to as "Information Groups (IGs)." A Type field identifies, and "tags" an Information Group. A tag helps PPGs discern appropriate action for unrecognized IGs provided by their children PGs. The Length field specifies the size of an IG, and the Value field is used for each IG for various different purposes. These packets (which contain only routing related information) are exchanged over special VCCs called Routing Control Channels (RCCs). RCCs are set up between all lowest level nodes that are either: (1) connected via physical links, or VPCs, or (2) between LGNs within the same PG. RCCs only transport routing related data.

Lowest level nodes that are physically adjacent establish a RCC VCC (Path 0, Channel 18) to exchange PNNI routing packets. LGNs within the same peer group establish RCC SVCCs (with no predefined VPI or VCI) to exchange routing packets. Lowest level nodes that are connected via a Virtual Path Connection (VPC), use the same VPI as the VPC, but with a VCI = 18.

RCCs use AAL-5 with a null SSCS (which means unassured mode information transfer). The PNNI routing protocol uses the CPCS (AAL-5 sublayer) error detection capability to recover from lost or corrupted AAL-SDUs.

After a RCC is established between logically adjacent nodes, they both activate the Hello process to confirm that they are in the same PG. Then they initiate a topology database exchange to synchronize their databases. Only after the topology synchronization is complete, is the link information pertaining to the new SVCC flooded.



PNNI Header

Packet Type 2 Bytes (B)	Packet Length 2B	Protocol Version 1B	Newest Version Supported 1B	Oldest Version Supported 1B	Reserved 1B

Packet Type Values
1 = Hello
2 = PTSP
3 = PTSP Ack
4 = Database Summary
5 = PTSE Request

PNNI Hello Packet

PNNI Header 8B	Flags 2B	Node ID (NID) 22B	ATM End System Address (AESA) 20B	Peer Group ID (PGID) 14B	Remote Node ID 22B

Port ID 4B	Remote Port ID 4B	Hello Interval 2B	Reserved 2B	Depends on what nodes are communicating. See Below

If communication is to an outside Peer Group then add the following to the PNNI Hello Packet

(1) Aggregation Token

Type "32" 2B	Length 2B	Aggregation Token 2B

(2) Nodal Hierarchy List

Type "33" 2B	Length 2B	Sequence Number 4B	Reserved 2B	Level Count 2B

Repeat Level Count Times

Next Higher Level Lobical Node ID 22B	Next Higher Level ATM End System Address 20B	Next Higher Level Peer Group ID 14B

(3) Uplink Information Attribute

Type "34" 2B	Length 2B	Sequence Number 4B	All Outgoing Resource Availability Information Groups (IG) (type =128), or optional IGs to describe the reverse direction of the uplink

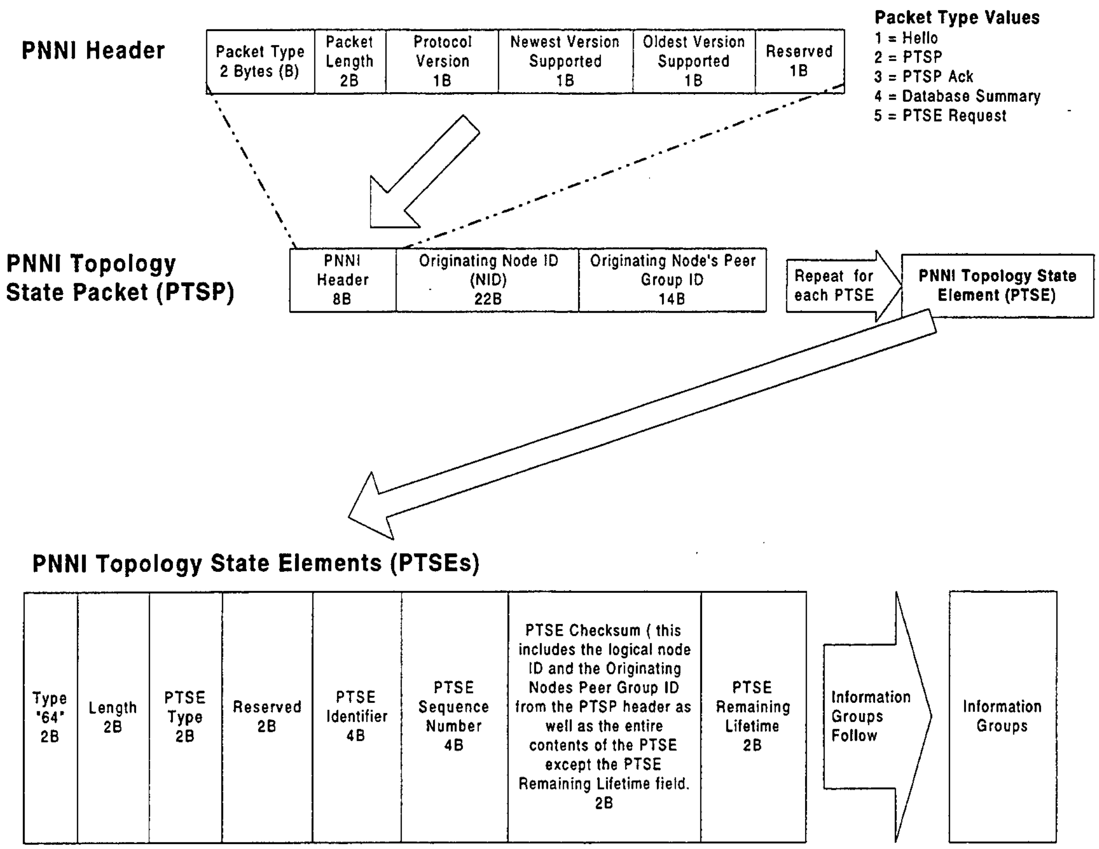
If communication is between Logical Group Nodes then add the following to the PNNI Hello Packet

Type "35" 2B	Length 2B	Reserved 2B	Horizontal Link Count 2B

Repeat Horizontal Link Count Times

Aggregation Token 4B	Local LGN port 4B	Remote LGN port 4B

Figure 28. PNNI Hello Packet



PNNI Header

Packet Type 2 Bytes (B)	Packet Length 2B	Protocol Version 1B	Newest Version Supported 1B	Oldest Version Supported 1B	Reserved 1B

Packet Type Values
1 = Hello
2 = PTSP
3 = PTSP Ack
4 = Database Summary
5 = PTSE Request

PNNI Topology State Packet (PTSP)

PNNI Header 8B	Originating Node ID (NID) 22B	Originating Node's Peer Group ID 14B

PNNI Topology State Elements (PTSEs)

Type "64" 2B	Length 2B	PTSE Type 2B	Reserved 2B	PTSE Identifier 4B	PTSE Sequence Number 4B	PTSE Checksum (this includes the logical node ID and the Originating Nodes Peer Group ID from the PTSP header as well as the entire contents of the PTSE except the PTSE Remaining Lifetime field. 2B	PTSE Remaining Lifetime 2B

Information Groups

Information Groups (IGs) specify exterior and interior reachability, addresses, aggregation information, other PTSEs, Generic Connection Admission Control (GCAC), nodal PTSE Acks and Summaries, system capabilities, as well as requests for PTSEs. IGs are used, not only in the PTSEs, but in many of the PNNI packets to transfer information between peer members, peer groups, or peer ancestors.

Figure 29. PNNI Topology State Packet (PTSP)

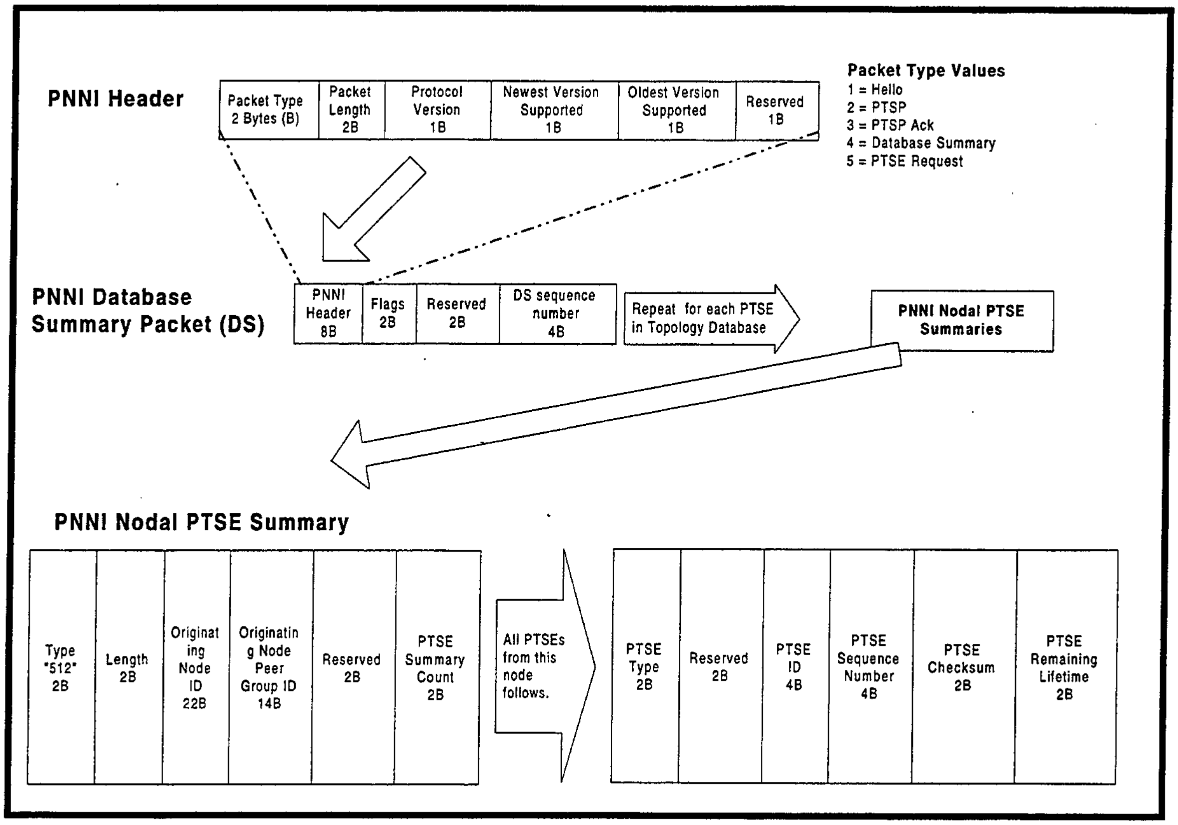


Figure 30. PNNI Database Summary (DS) Packet

(5) Path Selection Process. PNNI uses Source Routing to process connection requests, however it does not specify a required path selection algorithm. Source Routing in PNNI uses Designated Transit Lists (DTLs) to propagate a UNI SETUP request correctly. When a PNNI node within a PG receives a UNI SETUP request from a connected ATM End Station (AES), it determines if it has the capability to reach the destination with the desired traffic attributes. If it does then it sets up a complete path list (DTL) for the connection request to reach the desired destination, embeds this in a PNNI SETUP message and forwards it to the next node in-line to reach the destination. The complete path list contains detailed node by node transit information pertaining to its PG, and (if needed) summarized ancestor node information to reach the desired destination. If there are problems discovered en-route as a connection request propagates through a network using the path list, then a process called "Crankback" occurs to reroute the connection request around the problem area. Please See

(6) Figure 31, this is an edited version of Figure 26 to illustrate Path Selection, and Crankback.

Only the lowest level nodes pass user traffic. Lowest level nodes have detailed information about each nodes associated PG, as well as learned information about common ancestor PGs of physically connected outside neighbors. The main purpose of a PNNI hierarchy is to simplify reachability information so that routing tables will be small and efficiently searched. Higher level PGs and LGNs facilitate this process by delineating a method of organizing reachability information to lowest level nodes.

(7) DTL Example. ATM End Station (AES) A.3.3.100 in

(8) Figure 31 needs to set up a connection to AES C.2.200. I will call the source of the connection request A.3.3.100, "X," and the final destination of the connection request C.2.200, "Y."

(9) Figure 31 has bold circles numbered 1 through 11. Each numbered bold circle correspondingly represents the numbered steps described below for a connection request process. Note that onlya subset of the steps taken are described to illustrate how DTLs are used. The following example starts with the Source A.3.3.100 "X" sending a UNI SETUP message to node A.3.4.

1. A.3.4
   - Looks at its routing table which contains reachability information to the following summarized addresses [A.3.3, A.3.2, A.3.1, A.3, A.2, A.4, A.1, A, B, and C].
   - Picks a path that will provide desired transport attributes (delay, bit rate, etc).
   - Creates a Designated Transit List (DTL) to the last destination it knows of in its routing table that has the largest prefix match for C.2.200 "Y" (largest prefix match is node C). The DTL list is a stack, three levels high. The DTL list is as follows:

     DTL [A.3.4, A.3.1] offset 2 (offset points to next node to receive connection request)

     DTL [A.3, A.2, A.1] offset 1 (offset points to its Parent Peer Group node)

     DTL [A, B, C] offset 1 (offset points to its grandparent node, which is a common ancestor PG to Y)
   - Forwards a SETUP message with the embedded DTL to A.3.1.
2. A.3.1
   - Looks at the preceding DTL list, and determines that it is the last node on that list for PG A.3.
   - It removes the top DTL from the list, looks at the next PG (or node) it needs connectivity to (A.2), and then finds a directly connected node A.2 in its routing table.
   - It moves the offset on the highest remaining DTL to point to A.2 and forwards the SETUP message with the embedded DTL list to A.2. The DTL list looks like the following:

     DTL [A.3, A.2, A.1] offset 2 (offset points to the next node in that PG)

     DTL [A, B, C] offset 1 (offset points to its grandparent node in common ancestor PG with Y)
3. A.2
   - Looks at the DTL list, and since it isn't an LGN, it doesn't need to add more detailed routing information to the list.
   - It modes the offset pointer to the next node on the list, A.1 and passes the SETUP message with the embedded DTL list to A.1. The DTL list looks like the following:

DTL [A.3, A.2, A.1] offset 3 (offset points to the next node in that PG)

DTL [A, B, C] offset 1 (offset points to its grandparent node in common ancestor PG with Y)

4. A.1
   - Looks at preceding DTL list, and determines that it is the last node on that list for PG A.
   - It removes the top DTL from the list [looks at the next PG (or node)] it needs connectivity to (B), and then finds a directly connected node (B.1) in its routing table.
   - It moves the offset on the highest remaining DTL to point to B and forwards the SETUP message with the embedded DTL list to B.1. The DTL list looks like the following:

     DTL [A, B, C] offset 2 (offset points to node B en-route to C to reach Y)
5. B.1
   - Looks at the DTL list and determines that since it is the border node for PG B and is receiving the connection request it must designate a path through PG B to reach PG C.
   - It finds in its routing table many paths to reach PG C, and it selects one based on routing parameters.
   - Since it has to add another DTL to the existing DTL list, it makes a copy of the DTL list. The final DTL list it creates looks like the following:

     DTL [B.1, B.2] offset 2 (offset points to the next node in that PG)

     DTL [A, B, C] offset 2 (offset points to the current PG)
   - It sends the SETUP message with the embedded DTL to B.2

By this point the process of what DTLs are, and how they work should be fairly evident. The connection request continues through PG C to Y.

(10) Crankback. The term Crankback was used before, but not fully described. Using

(11) Figure 31 to illustrate what happens, suppose that the link between C.3 and C.2 became faulty (a backhoe dug it up).

If C.4 did not know about the faulty C.3-C.2 link when it created the detailed DTL pertaining to PG C, then it would receive a RELEASE message containing a Crankback information element from C.3, after it send the connection request to it. The RELEASE message it receives indicates a Crankback level 96. Level 96 is the level of peer group C.

It is important to note here that when C.4 first received the DTL list from B.2, it created the top detailed DTL with a level of 96 (it's level). Since C.4 added the detailed PG C DTL with Crankback level 96, and the Crankback level in the RELEASE message pertaining to the Crankback is not higher than the level it created, it creates a new one (this time sending it via C.1 rather than C.3 – note many options for path selection are possible).

This process of cranking allows the network to try alternate routes to a destination (closer to the problem area) without having to send the connection request all the way back to the originating PNNI node. If a fault occurs, and C.4 cannot process another path to the destination, after receiving a release then it will change the Crankback level to its parent PG (level 80) and send a RELEASE message back to node B.2.

When a node receives a RELEASE with a Crankback level associated with a DTL that it created, it will look for alternative paths through its PG only if the Crankback level number returned is equal to, or higher than what it was when the node created the DTL.

If the Crankback level returned in a RELEASE message is lower than the level number it assigned to the DTL when it created it, it will Crankback the connection path even further.

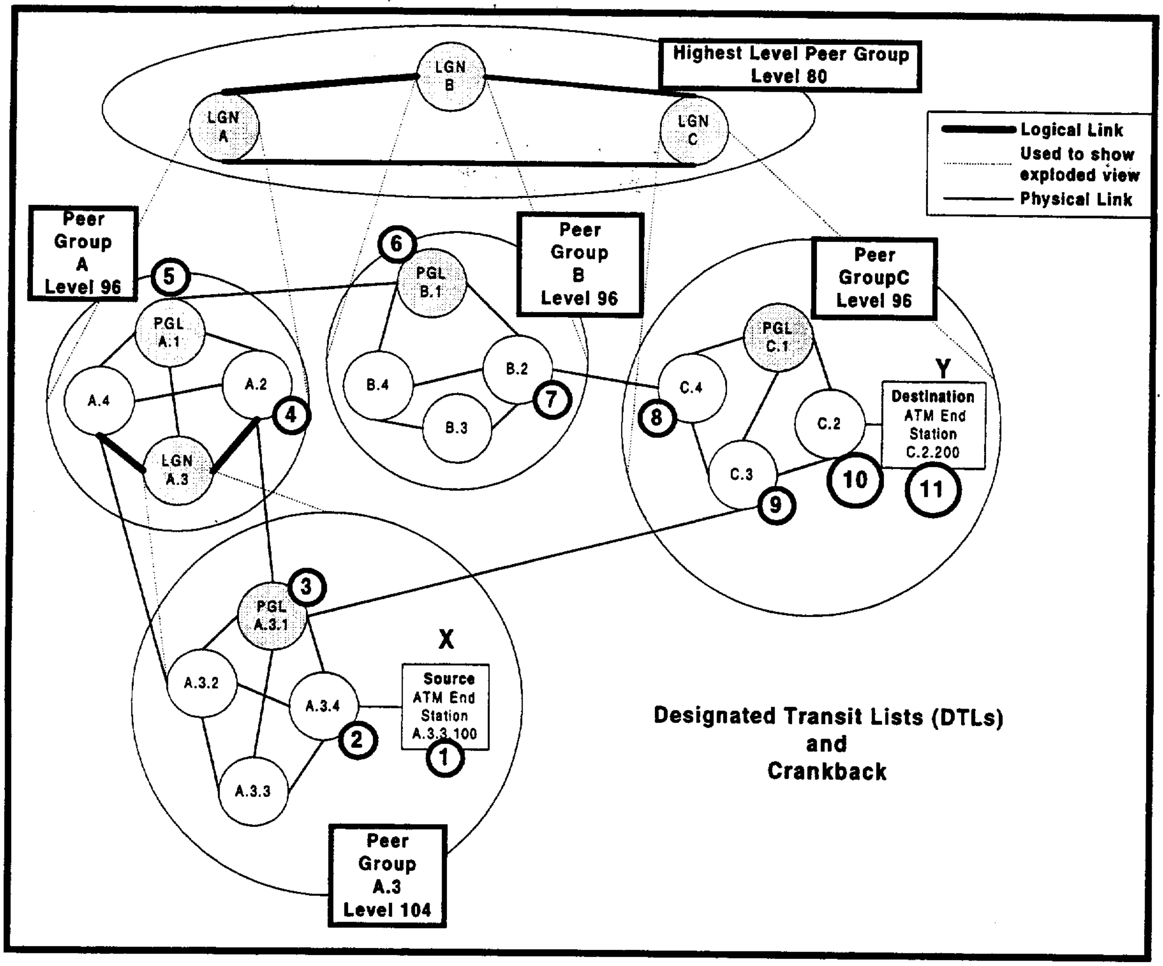


Figure 31. PNNI – DTLs and Crankback Example

b. *PNNI Signaling*

PNNI V1.0 is based on the UNI 4.0/Q.2931 Signaling protocol (with added functionality required for routing), as well as the Frame Relay NNI signaling (ITU-T Q.934). The signaling procedures between PNNI, UNI4.0, UNI3.1, and Q.2931are very similar.

In the PNNI two types of signaling are specified, Associated Signaling, and Non-Associated Signaling. Associated signaling refers to VPCs where the layer 3 entities choose the VCs within the VPC that contains its signaling VC (only used for logical link VPCs). Non-associated signaling controls all VCs, and VPs on a physical link, except those associated with associated signaling. VPI/VCI 0/5 is the only non-associated signaling path/channel used in PNNI.

The PNNI specification is written in terms of a preceding (or originating) side of a PNNI connection, and a succeeding (or terminating) side of a PNNI connection (See Figure 8). At each end of a PNNI connection, state machines are used for call and connection control. The states associated with PNNI point-to-point calls are as follows (exactly as specified PNNI Section 6.2):

1. Null - No call exists.
2. Call Initiated - This state exists on the succeeding side if it received a call establishment request form the preceding network node but has not yet responded.
3. Call Proceeding Sent - This state exists when a succeeding side has acknowledged the receipt of the information necessary to establish a call.
4. Alerting Delivered - This state exists when a succeeding side has sent an ALERTING message to the preceding side.
5. Call Present - This state exists at a preceding side after it has sent a call establishment request to the succeeding side but has not yet received a response.
6. Alerting Received - This state exists when a preceding side has received an ALERRTING message from the succeeding side of the PNNI interface.
7. Call Proceeding Received - This state exists when a preceding side has received acknowledgment that the succeeding side has received the call establishment request.
8. Active - This state exists when the ATM connection has been established.
9. Release Request - This state exists when a network node has sent a request to the network node at the other side of the PNNI interface to release the connection and is waiting for a response.
10. Release Indication - This state exists when a network node has received a request from the network node at the other side of the PNNI interface to release the ATM connection and has not responded yet.
11. Other states - States associated with global call references as specified by UNI4.0/Q.2931 (all zero's) are not discussed in this document.

Machine state transitions are dependent upon time, as well as the type and content of PNNI messages sent, or received. The messages associated with PNNI call and connection control for point-to-point calls are as follows (exactly as specified in PNNI Section 6.3):

1. Call Establishment Messages
    - ALERTING. - This message is sent by the Succeeding side to indicate that called user alerting has been initiated.
    - CALL PROCEEDING. - This message is sent by the Succeeding side to indicate that the requested call/connection establishment has been initiated and no more call establishment information will be accepted.
    - CONNECT. - This message is sent by the Succeeding side and delivered to the Preceding side to indicate call/connection acceptance by the called user.
    - SETUP. - This message is sent by the Preceding side to the Succeeding side to initiate a call/connection establishment.
2. Call Clearing Messages
    - RELEASE.- This message is sent by a network node to an adjacent network node to indicate that it has cleared the connection and is waiting to release the call reference.

- RELEASE COMPLETE. - This message is sent by a network node to an adjacent node to indicate that it has cleared internally the connection (if any) and released the call reference.

3. Miscellaneous messages.
   - NOTIFY - This message is sent by either side to indicate information pertaining to a call/connection.
   - STATUS - This message is sent by either side in response to a STATUS ENQUIRY message or at any time to report certain error conditions.
   - STATUS ENQUIRY - This message may be sent by either side at any time to solicit a STATUS message for the peer entity. Sending a STATUS message in response to a STATUS ENQUIRY message is mandatory.
4. Messages added for the support of 64kibits/s based ISDN circuit mode services.
   - RESTART. - This message is sent by either side to request the recipient to restart (i.e., release all resources associated with) the indicated virtual channel/path or all virtual channels/paths controlled by the signaling virtual channel.
   - RESTART ACKNOWLEDGE - This message is sent to acknowledge the receipt of a RESTART message and to indicate that the requested restart is complete.
   - Other messages. - Are listed, however they will not be covered in this report.

Figure 35 and Figure 36, are provided as a description of the Call/Connection/Release procedures as described in the specification; note that these figures do not show every procedure associated with PNNI, and show the states of individual calls, not the interface. Timers are used to transition states in the event of failures. The list of timers as specified in the PNNI standard (PNNI refers to Table 7-1/Q.2931) is as follows:

1. T301
   - Time: A minimum of three minutes
   - Cause for Start: ALERTING Received
   - Cause for normal stop: CONNECT Received
   - Action at first expiry: Clear call
2. T303
   - Time: Four seconds
   - Cause for Start: SETUP sent
   - Cause for normal stop: ALERTING, CONNECT, RELEASE COMPLETE, or CALL PROCEEDING Received
   - Action at first expiry: Resend SETUP, restart T303
   - Action at second expiry: Clear network connection, enter the Null state
3. T308
   - Time: 30 seconds
   - Cause for Start: RELEASE sent

- Cause for normal stop: RELEASE, or RELEASE COMPLETE received
- Action at first expiry: Resend RELEASE, restart T308
- Action at second expiry: Place VC in maintenance condition, Release Call Reference, and enter Null state.

4. T309
    - Time: Ten seconds
    - Cause for Start: SAAL disconnection. Calls in stable states are not lost.
    - Cause for normal stop: SAAL reconnected
    - Action at first expiry: Clear network connection; release connection and call reference
5. T310
    - Time: PNNI only: 30 - 110 seconds, Q.2931: Ten seconds
    - Cause for Start: CALL PROCEEDING received
    - Cause for normal stop: ALERTING, CONNECT, RELEASE received
    - Action at first expiry: Clear call
6. T316
    - Time: 120 seconds
    - Cause for Start: RESTART sent
    - Cause for normal stop: RESTART ACKNOWLEGE received
    - Action at first expiry: RESTART may be sent several times
7. T317
    - Time: Implementation dependent, usually less than T316
    - Cause for Start: RESTART received
    - Cause for normal stop: Internal clearing of call references
    - Action at first expiry: Maintenance notification
8. T322
    - Time: Four seconds
    - Cause for Start: STATUS ENQUIRY sent
    - Cause for normal stop: STATUS, RELEASE, or RELEASE COMPLETE Received
    - Action at first expiry: STATUS ENQUIRY may be resent several times

(1) PNNI Signaling Message Encapsulation. PNNI signaling uses the SAAL assured mode service (as previously described) to help provide reliable message delivery. Figure 32 is an encapsulation example that shows how a PNNI SETUP message is encapsulated as it moves down the ATM associated layer structure.

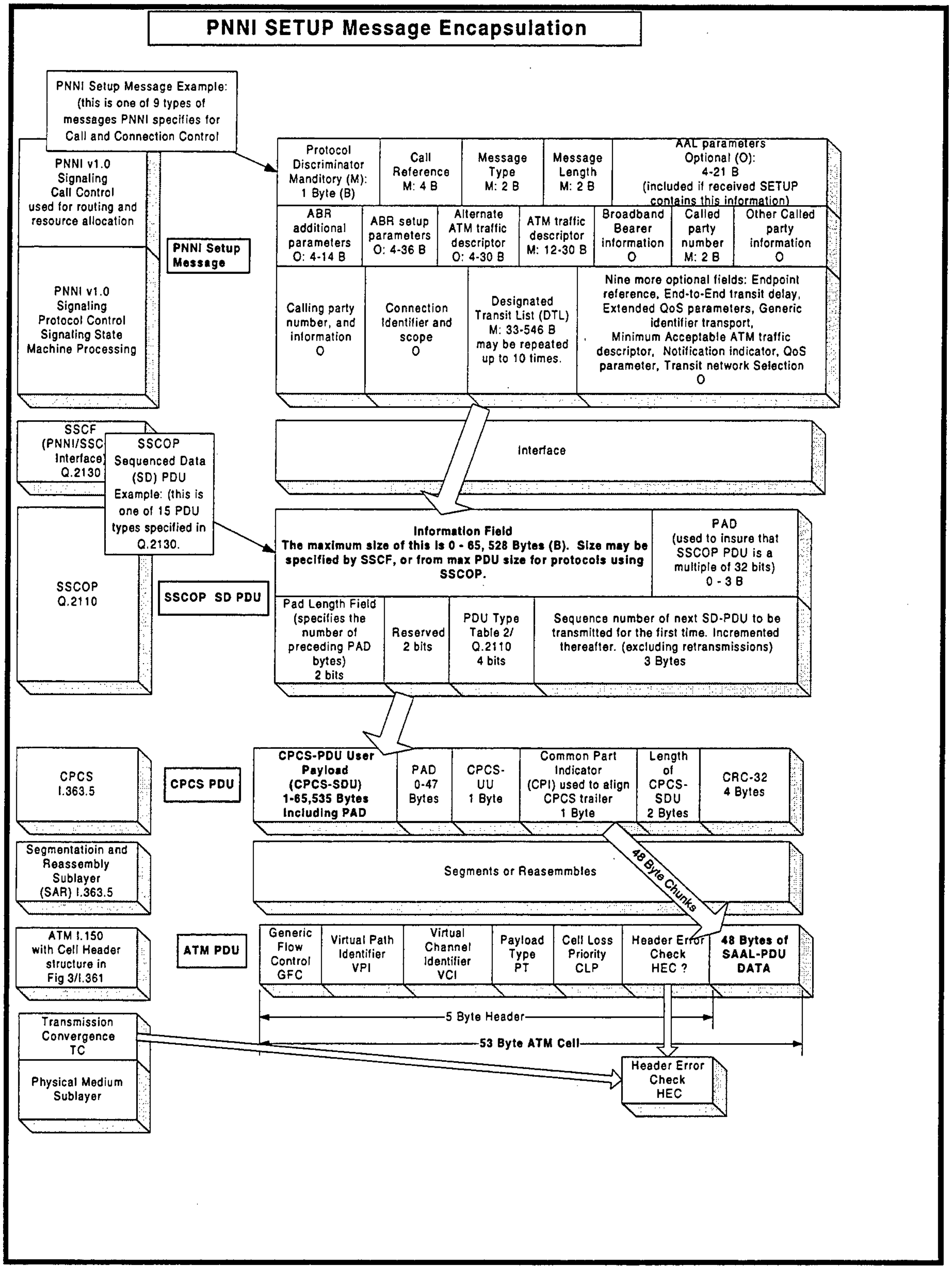

PNNI SETUP Message Encapsulation
PNNI Setup Message Example: (this is one of 9 types of messages PNNI specifies for Call and Connection Control
PNNI v1.0 Signaling Call Control used for routing and resource allocation
PNNI v1.0 Signaling Protocol Control Signaling State Machine Processing
PNNI Setup Message
Protocol Discriminator Manditory (M): 1 Byte (B)
Call Reference M: 4 B
Message Type M: 2 B
Message Length M: 2 B
AAL parameters Optional (O): 4-21 B (included if received SETUP contains this information)
ABR additional parameters O: 4-14 B
ABR setup parameters O: 4-36 B
Alternate ATM traffic descriptor O: 4-30 B
ATM traffic descriptor M: 12-30 B
Broadband Bearer information O
Called party number M: 2 B
Other Called party information O
Calling party number, and information O
Connection Identifier and scope O
Designated Transit List (DTL) M: 33-546 B may be repeated up to 10 times.
Nine more optional fields: Endpoint reference, End-to-End transit delay, Extended QoS parameters, Generic identifier transport, Minimum Acceptable ATM traffic descriptor, Notification indicator, QoS parameter, Transit network Selection O
SSCF (PNNI/SSCOP Interface) Q.2130
SSCOP Sequenced Data (SD) PDU Example: (this is one of 15 PDU types specified in Q.2130.
Interface
SSCOP Q.2110
SSCOP SD PDU
Information Field The maximum size of this is 0 - 65, 528 Bytes (B). Size may be specified by SSCF, or from max PDU size for protocols using SSCOP.
PAD (used to insure that SSCOP PDU is a multiple of 32 bits) 0 - 3 B
Pad Length Field (specifies the number of preceding PAD bytes) 2 bits
Reserved 2 bits
PDU Type Table 2/ Q.2110 4 bits
Sequence number of next SD-PDU to be transmitted for the first time. Incremented thereafter. (excluding retransmissions) 3 Bytes
CPCS I.363.5
CPCS PDU
CPCS-PDU User Payload (CPCS-SDU) 1-65,535 Bytes Including PAD
PAD 0-47 Bytes
CPCS-UU 1 Byte
Common Part Indicator (CPI) used to align CPCS trailer 1 Byte
Length of CPCS-SDU 2 Bytes
CRC-32 4 Bytes
Segmentatioin and Reassembly Sublayer (SAR) I.363.5
Segments or Reasemmbles
48 Byte Chunks
ATM I.150 with Cell Header structure in Fig 3/I.361
ATM PDU
Generic Flow Control GFC
Virtual Path Identifier VPI
Virtual Channel Identifier VCI
Payload Type PT
Cell Loss Priority CLP
Header Error Check HEC ?
48 Bytes of SAAL-PDU DATA
5 Byte Header
53 Byte ATM Cell
Transmission Convergence TC
Physical Medium Sublayer
Header Error Check HEC

Figure 32. Control Plane Message Encapsulation and Layers

(2) PNNI and UNI 3.1 Signaling Messages. All PNNI signaling message have five common Parts (See Figure 33), which have special emphasis with regard to PNNI error detection: (1) Protocol Discriminator (1 Byte), (2) Call Reference (4 Bytes), (3) Message Type (2 Bytes), and (4) Message Length (2 Bytes), (5) General Information Elements.

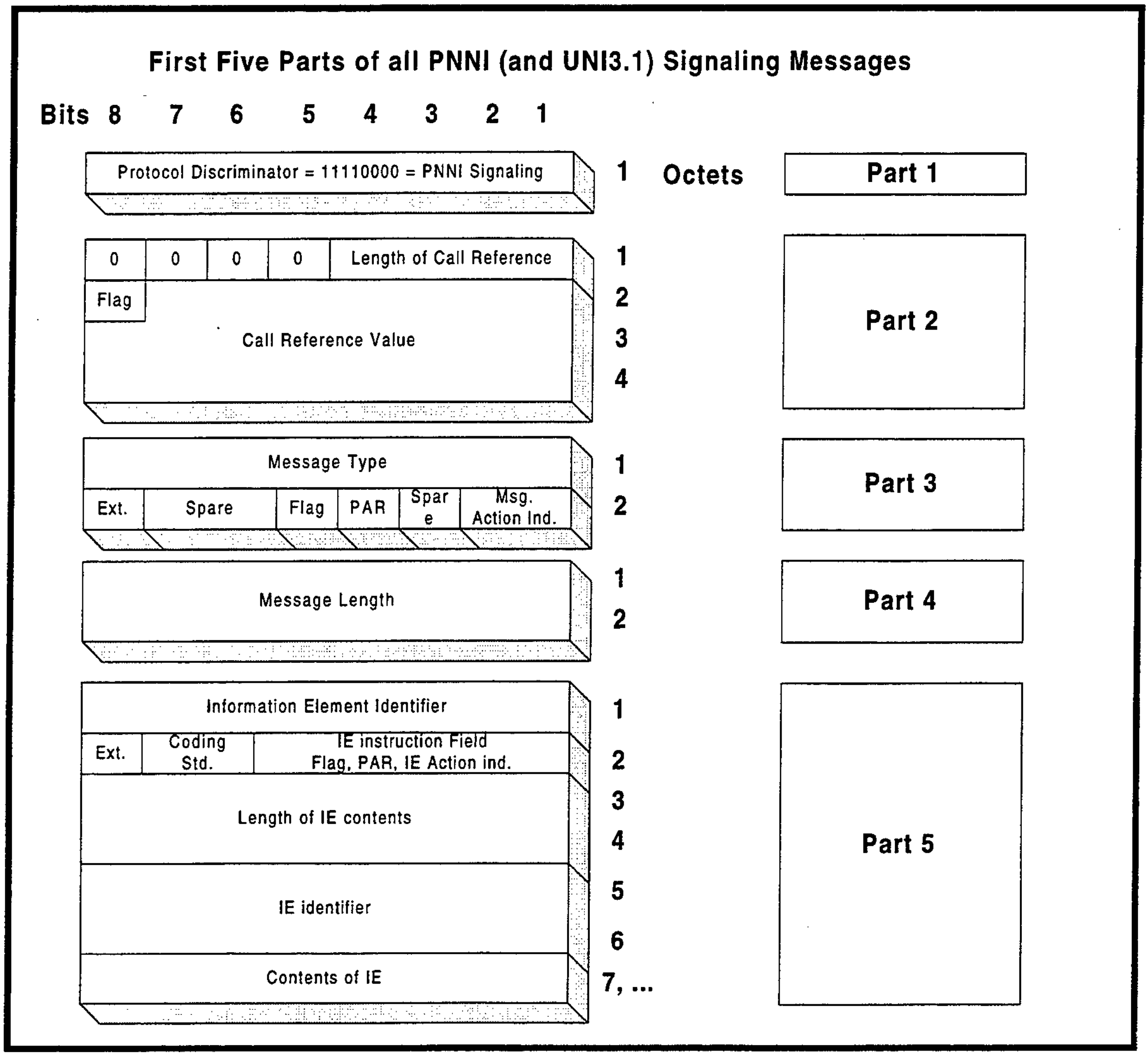


Figure 33. PNNI/UNI/Q.2931 - The First Five Parts of Every PNNI, UNI, and Q.2931 Signaling Message

Figure 33 is common with PNNI, UNI 3.1, and Q.2931. The Standards' Section References are provided. Part 1 (shown in Figure 33) is the Protocol Discriminator. The Protocol Discriminator distinguishes between message types such as: (1) a PNNI signaling message, (2) Q.2931 = 00001001 (UNI 3.1 uses this), and (3) other standards based messages. The following rules apply for Part 1:

1. For PNNI if a Protocol Discriminator is coded as anything other than 11110000, the entire message is ignored.
2. If a Protocol Discriminator is coded as anything other than 00001001, then UNI3.1, and Q.2931 will ignore it.

Part 2 is the Call Reference. The originating side of an interface assigns the Call Reference Value, which does not have end-to-end significance. For UNI3.1 the Call Reference is assigned by the network side of an interface. Call Reference values are given in terms of Virtual Path Connection Identifier (VPCI) and Virtual Channel Identifiers (VCI). A VPCI is the same as a VPI for signaling channels that control paths on only one interface. If a signaling channel controls paths on more than one interface, the VPCI is composed of not only the VPI, but also an Interface identification number. Two identical Call Reference values may be associated with a single interface, but they must originate at different ends of a link (the Call Reference flag identifies which end of the link originates each call). The originator of a Call Reference value must set the Call Reference flag equal to "0." Calls sent by an interface that did not originate the Call Reference value must set the Call Reference flag equal to "1." Call Reference errors associated with STATUS or STATUS ENQUIRY result in either the call being cleared, ignored, or returned into an active state via various types of corrective actions. If the first four bits of the first byte in Part 2 contain values that indicate a Call Reference value length other than 3 bytes the entire message is ignored. The message is also ignored if the last 4 bits of the first byte in Part 2 are something other than "0000."

For messages other than SETUP, RELEASE COMPLETE, STATUS ENQUIRY, and STATUS, Call References are checked, and clearing is initiated (by sending a RELEASE COMPLETE with CAUSE "invalid call reference value") if the call is not recognized as an active call or a call-in-progress. RELEASE COMPLETE messages with invalid Call Reference Values are ignored.

If a received SETUP message contains a Call Reference Flag equal to "1" and a Call Reference value that is not recognized the message is ignored. If the received SETUP message contains a Call Reference flag equal to "0" and a recognized receiving-end-originated active call or call-in-progress Call Reference Value the message is ignored.

A RESTART, RESTART ACK, and STATUS must not have Call Reference value set to all zeros. If it is, then a STATUS is returned with cause "invalid call reference" and the state associated with the current global call reference.

STATUS messages returned with invalid Call Reference values follow the Status Enquiry procedure in Figure 34. If a received STATUS message indicates that a call is not in the Null state, but on the receiver it is in the Null state, then a RELEASE COMPLETE message is returned. If a received STATUS message indicates that a call is in the Null state, but on the receiver it is not in the Null state, then the call resources are released, and the call is moved into the null state (for PNNI, Call Control is notified). If a received STATUS message contains no anomalies with respect to the local state then nothing is done. Other errors are implementation specific.

Part 3 is the Message Type. The first field identifies the type of message. Examples include CONNECT, SETUP, RELEASE, RESTART, STATUS, etc. See Tables 4-2/Q.2931, UNI3.1 Section 5.4.4.1, and PNNI V1.0 Section 6.4.4.1 for complete lists of Q.2931, UNI3.1, and PNNI V1.0 values, respectively associated with this field. The Flag field is used to indicate whether the Message Action Indicator field should be used in the event of an unrecognized message. The Pass Along Request (PAR) field is used to tell a destination PNNI interface whether to process the message field and check for errors, or pass the message to the next PNNI interface without checking for message type compatibility or errors. Note that for UNI3.1, the PNNI Part 3 PAR field is not used; it is called a "spare" field instead. The Message Action Indicator can specify one of the following four actions for PNNI, UNI3.1, or Q.2931: (1) clear call, (2) discard and ignore, (3) discard and report status, and (4) reserved.

If the Flag field in Part 5 indicates that the IE Instruction field should not be used, then the following rules associated with message type or message sequence errors apply:

1. Whenever an unexpected message is received except a RELEASE or RELEASE COMPLETE message and the state of the message is something other than the Null State the message is ignored and a STATUS message is returned with cause.
2. Whenever an unrecognized message is received the message is ignored and a STATUS message is returned with cause. The receipt of an unexpected RELEASE message in response to a SETUP will result in the following: (1) the release of the VC, (2) the clearing of the connection, (3)the return of a RELEASE COMPLETE message, (4) the release of the call reference, (5) all timers stopped, (6) the call entering the Null state, and (7) PNNI call control notified.
3. The receipt of an unexpected RELEASE COMPLETE message will result in the following: (1) the release of the VC, (2) the clearing of the connection, (3) the release of the call reference, (4) all timers stopped, (5) the call entering the Null state, and (6) PNNI call control notified.
4. If the message does not contain the full Message Length IE field it will be ignored.
5. If the Message Length value is not equal to that received then whatever part of the message that can be processed is.processed. An exception to this is if mandatory IEs or mandatory IE content is missing or in error (See Part 5 discussion).

Part 5 is the Information Elements (IE) of a message. A message may contain many IEs. Some IEs must be included within a message type, these are called "mandatory information elements"; all others are called "non-mandatory informational elements." Thirty-nine IE types are specified in Table 6-5/PNNI V1.0, twenty-six are specified in Table 4-3/Q.2931, and twenty-one are specified in Table 5-5/UNI3.1. Example IEs include Calling Party Number, Called Party Number, Connection Scope Selection, ABR Setup Parameters, Connection Identifier, and End-to-End Transit Delay. The Coding Standard field identifies a coding standard that is used for the rest of the message. A Coding may be an ITU-T, ISO/IEC, a national standard, or another standards as long as they are also present on a destination interface of a link. The IE Instruction Field has a Flag, Pass Along Request Field (PAR), and an IE Action Indication field. The Flag field is used to indicate whether or not the IE Instruction field should be used in the event of an unrecognized IE Identifier, unrecognized IE content, or unexpected IE. The Pass Along Request (PAR) field is used to tell the destination PNNI interface whether or not to process the IE Action Indicator field and check for errors or pass the message to the next PNNI interface without checking for IE type compatibility or errors. The IE Action Indicator can specify one of the following four actions for PNNI, UNI3.1, or Q.2931 signaling messages: (1) clear call, (2) discard IE, and proceed, (3) discard IE, proceed, and report status, (4) discard message, and ignore, and (5) discard message, and report status.

If any message except SETUP, RELEASE, or RELEASE COMPLETE have missing mandatory IEs the message is ignored and a STATUS is returned with cause. If a SETUP is received without a mandatory IE a RELEASE COMPLETE is returned with a cause. If a RELEASE is received without a CAUSE IE then clearing is initiated with a RELEASE COMPLETE containing a CAUSE IE returned. If a RELEASE COMPLETE is received without a CAUSE IE then clearing is initiated with a returned RELEASE COMPLETE containing cause.

If the Flag field in Part 5 indicates that the IE Instruction field should not be used then the following rules associated with mandatory IE content apply:

1. If any message except SETUP, RELEASE, or RELEASE COMPLETE have mandatory IEs with invalid content the message is ignored, and a STATUS is returned with a Cause IE.
2. If a SETUP is received with a mandatory IE with invalid content , a RELEASE COMPLETE is returned with a cause.
3. If a RELEASE is received without a valid Cause IE content, then clearing is initiated, with a RELEASE COMPLETE containing a Cause IE, returned.
4. If a RELEASE COMPLETE is received without a valid Cause IE content then clearing is initiated with a RELEASE COMPLETE containing cause. Mandatory IEs, which exceed maximum lengths are treated as if they have content error.

If the Flag field in Part 5 indicates that the IE Instruction field should not be used, then the following rules associated with non-mandatory IEs apply:

1. All recognized messages which have unrecognized IEs process only those IEs that are recognized and have valid content.
2. The receipt of any message except a RELEASE, or RELEASE COMPLETE message with unrecognized IEs, as an option, in addition to what was previously described, must return a STATUS message with one Cause IE which contains the call state of the receiver after processing the valid IEs, and optionally, if the Cause IE has a diagnostic field, diagnostic information pertaining to the failed IEs.
3. If a RELEASE message with an unrecognized IE is received, a RELEASE COMPLETE message (instead of an optional STATUS message) is returned with a Cause IE as well diagnostic information if the Cause IE has the diagnostic field present.
4. If a RELEASE COMPLETE message is received with unrecognized IEs then the unrecognized IEs are not processed and nothing is returned to the sender. The PNNI standard specifies two additional rules that are followed if Broadband-locking shifting, or Broadband-non-locking shift IEs are within a message (these is not included in this thesis).

If the Flag field in Part 5 indicates that the IE Instruction field should not be used then the following rules associated with content error in non-mandatory IEs with apply:

1. All recognized messages, which have recognized IEs with invalid content process only those IEs that are recognized and have valid content.
2. A STATUS message may be returned with one Cause IE, which contains the call state of the receiver after processing the valid IEs, or optionally, if the Cause IE diagnostic field is present, diagnostic information pertaining to the failed IEs. IEs exceeding the maximum length are treated as IEs with invalid content.

If the Flag field in Part 5 indicates that the IE Instruction field should not be used then the following rule associated with incorrectly placed IEs apply: If a message is received with a valid IE, but the IE is not defined to be in the message, then the IE is treated as an unrecognized IE in the message.

When a signaling entity receives a SAAL reset (by the SAAL layer), all calls being cleared are cleared, all calls active, and those being established are maintained. A STATUS enquiry procedure is performed for calls in the active state, and optionally performed for calls in the Establishment State.

When a signaling entity receives a SAAL Release primitive indicating that there is a SAAL disconnection, all calls that are not in the active state are cleared. See Figure 34. If any calls are not in the active state then the signaling layer will try to re-establish the SAAL connection, then perform a Status Enquiry procedure for each call/connection to insure that the call states at the peer SAAL are correct. Note that the SAAL Release primitive described in this section is not the same as a connection RELEASE, shown in Figure 22, Figure 23, and Figure 24. Note. 6.11 info is in the figure.

If a SAAL connection cannot be established within a default time of 10 seconds, the connection will be cleared, Call References and VCs are released, and the network connection is disconnected (if applicable). The action taken by a signaling entity for each call state that is in error (as per the received STATUS message) is an implementation dependent decision unless otherwise noted in this thesis. Note that the default time for the return of call state information within a STATUS message is 4 seconds. PNNI only: Call control is notified if a connection cannot be re-established, and of each call failure.

If the Flag and (PNNI only: PAR) fields in Parts 3 and/or 5 indicate that in the event of unrecognized or unexpected message reception explicit actions (as indicated in the Action Ind. Fields of Parts 3 and 5) should be followed then the following rule applies:

1. The Message Action Indicator can specify one of the following four actions for PNNI, UNI3.1, or Q.2931: (1) clear call, (2) discard and ignore, (3) discard and report status, and (4) reserved.
2. The IE Action Indicator can specify one of the following four actions for PNNI, UNI3.1, or Q.2931 signaling messages: (1) clear call, (2) discard IE, and proceed, (3) discard IE, proceed, and report status, (4) discard message, and ignore, and (5) discard message, and report status.

Messages, originating from the PNNI Call Control layer are processed by state machine programs in the PNNI Protocol Control layer. State machine diagrams for Call/Connection/Release at the Preceding side and Succeeding sides of a PNNI link are shown in Figure 35 and Figure 36 respectively. Note that the PNNI state machine diagrams closely resemble those of the UNI (Figure 22, Figure 23, Figure 24, and Figure 25): both are based on the Q.2931 standard.

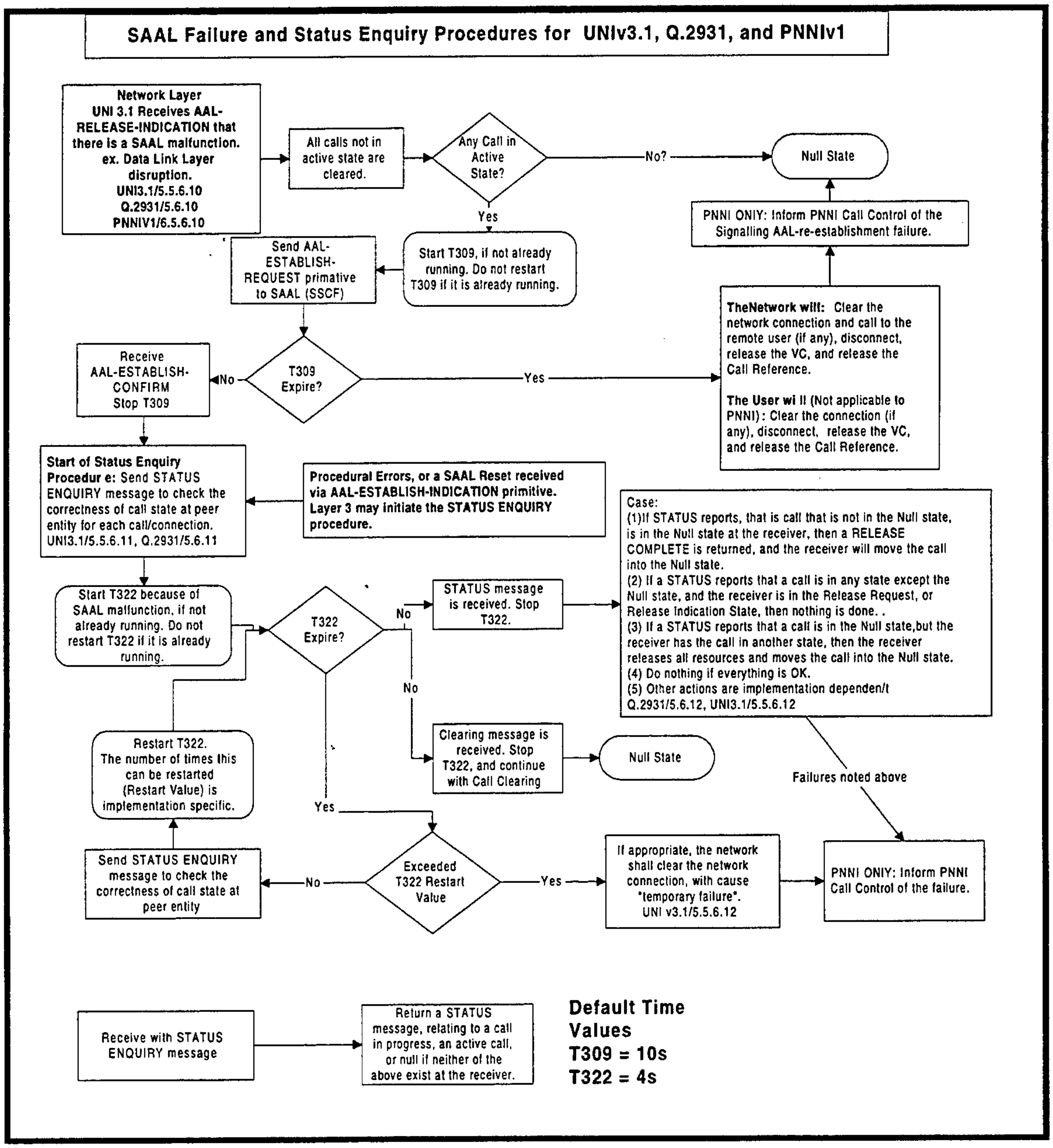

SAAL Failure and Status Enquiry Procedures for UNIv3.1, Q.2931, and PNNIv1
Network Layer UNI 3.1 Receives AAL-RELEASE-INDICATION that there is a SAAL malfunction. ex. Data Link Layer disruption. UNI3.1/5.5.6.10 Q.2931/5.6.10 PNNIV1/6.5.6.10
All calls not in active state are cleared.
Any Call in Active State?
No?
Null State
Yes
PNNI ONIY: Inform PNNI Call Control of the Signalling AAL-re-establishment failure.
Start T309, if not already running. Do not restart T309 if it is already running.
Send AAL-ESTABLISH-REQUEST primative to SAAL (SSCF)
TheNetwork will: Clear the network connection and call to the remote user (if any), disconnect, release the VC, and release the Call Reference.
The User wi ll (Not applicable to PNNI) : Clear the connection (if any), disconnect, release the VC, and release the Call Reference.
Receive AAL-ESTABLISH-CONFIRM Stop T309
No
T309 Expire?
Yes
Start of Status Enquiry Procedur e: Send STATUS ENQUIRY message to check the correctness of call state at peer entity for each call/connection. UNI3.1/5.5.6.11, Q.2931/5.6.11
Procedural Errors, or a SAAL Reset received via AAL-ESTABLISH-INDICATION primitive. Layer 3 may initiate the STATUS ENQUIRY procedure.
Case:
(1)If STATUS reports, that is call that is not in the Null state, is in the Null state at the receiver, then a RELEASE COMPLETE is returned, and the receiver will move the call into the Null state.
(2) If a STATUS reports that a call is in any state except the Null state, and the receiver is in the Release Request, or Release Indication State, then nothing is done. .
(3) If a STATUS reports that a call is in the Null state,but the receiver has the call in another state, then the receiver releases all resources and moves the call into the Null state.
(4) Do nothing if everything is OK.
(5) Other actions are implementation dependen/t
Q.2931/5.6.12, UNI3.1/5.5.6.12
Start T322 because of SAAL malfunction, if not already running. Do not restart T322 if it is already running.
T322 Expire?
No
STATUS message is received. Stop T322.
No
Restart T322. The number of times this can be restarted (Restart Value) is implementation specific.
Clearing message is received. Stop T322, and continue with Call Clearing
Null State
Failures noted above
Yes
Send STATUS ENQUIRY message to check the correctness of call state at peer entity
No
Exceeded T322 Restart Value
Yes
If appropriate, the network shall clear the network connection, with cause "temporary failure". UNI v3.1/5.5.6.12
PNNI ONIY: Inform PNNI Call Control of the failure.
Receive with STATUS ENQUIRY message
Return a STATUS message, relating to a call in progress, an active call, or null if neither of the above exist at the receiver.
Default Time Values
T309 = 10s
T322 = 4s

Figure 34. SAAL Failure and Status Enquiry Procedures for UNI3.1, Q.2931, and PNNI

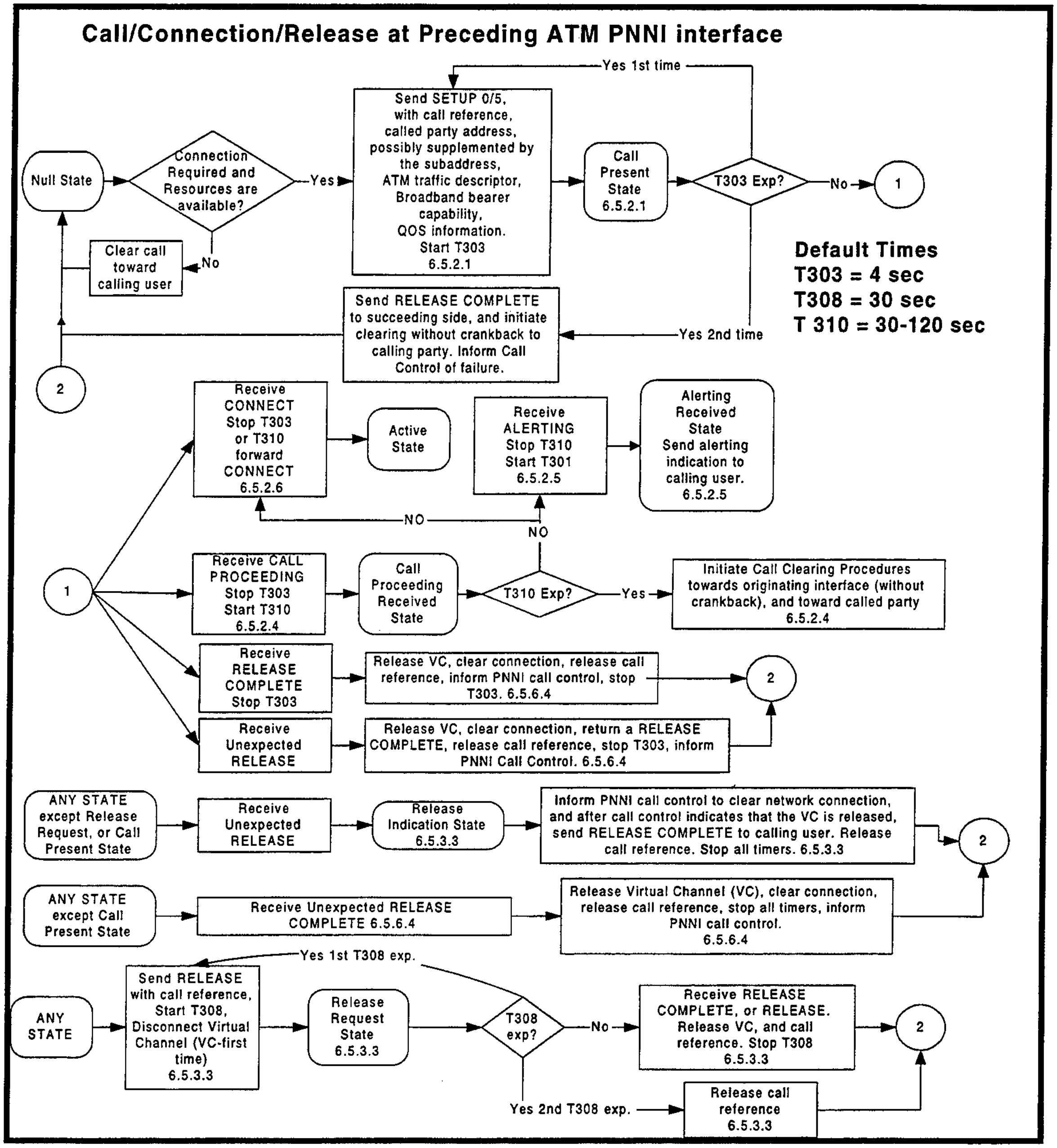
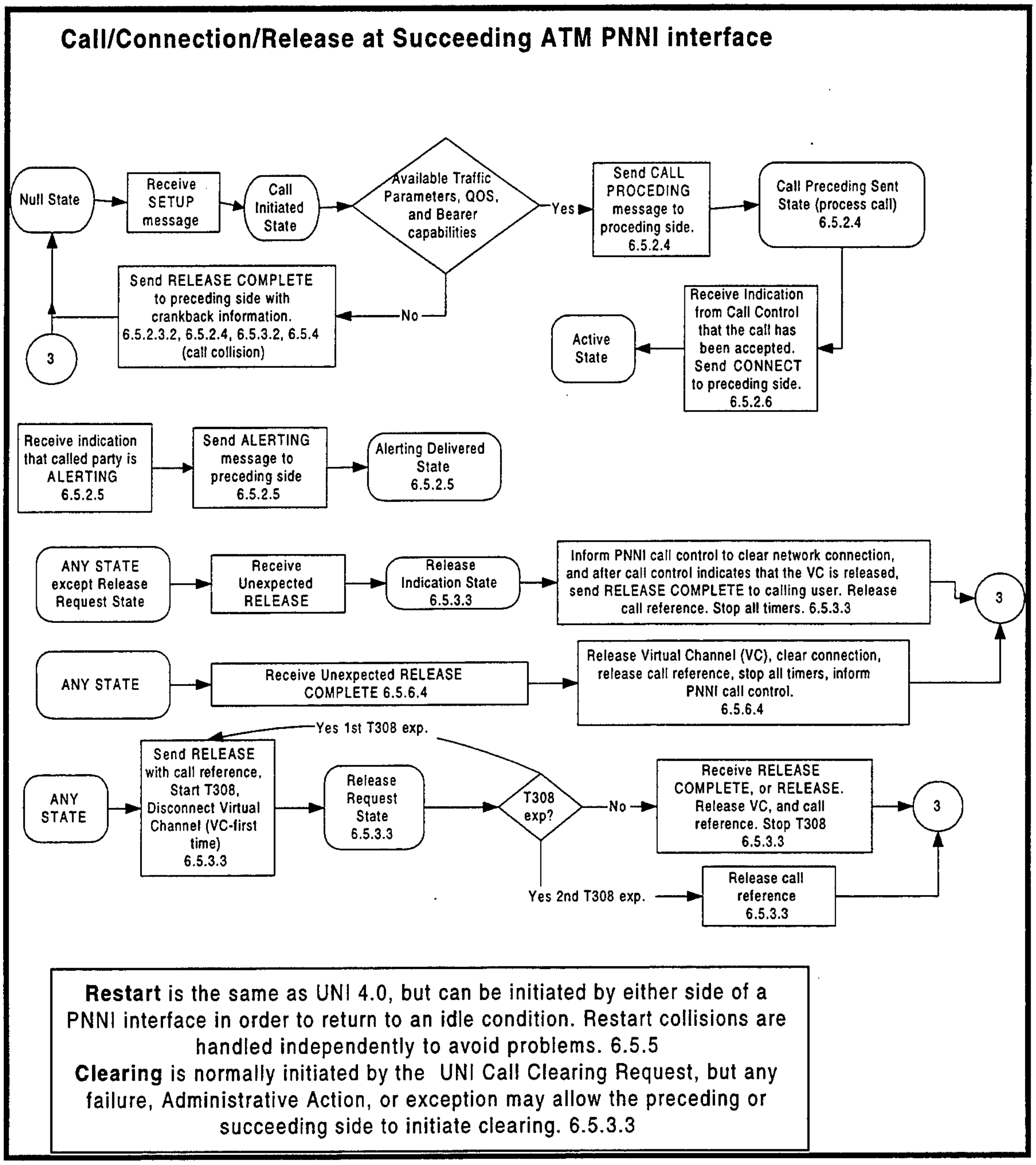

Call/Connection/Release at Preceding ATM PNNI interface
Yes 1st time
Null State
Connection Required and Resources are available?
Yes
No
Send SETUP 0/5, with call reference, called party address, possibly supplemented by the subaddress, ATM traffic descriptor, Broadband bearer capability, QOS information. Start T303 6.5.2.1
Call Present State 6.5.2.1
T303 Exp?
No
1
Clear call toward calling user
Default Times
T303 = 4 sec
T308 = 30 sec
T 310 = 30-120 sec
Send RELEASE COMPLETE to succeeding side, and initiate clearing without crankback to calling party. Inform Call Control of failure.
Yes 2nd time
2
Receive CONNECT Stop T303 or T310 forward CONNECT 6.5.2.6
Active State
Receive ALERTING Stop T310 Start T301 6.5.2.5
Alerting Received State Send alerting indication to calling user. 6.5.2.5
NO
NO
1
Receive CALL PROCEEDING Stop T303 Start T310 6.5.2.4
Call Proceeding Received State
T310 Exp?
Yes
Initiate Call Clearing Procedures towards originating interface (without crankback), and toward called party 6.5.2.4
Receive RELEASE COMPLETE Stop T303
Release VC, clear connection, release call reference, inform PNNI call control, stop T303. 6.5.6.4
2
Receive Unexpected RELEASE
Release VC, clear connection, return a RELEASE COMPLETE, release call reference, stop T303, inform PNNI Call Control. 6.5.6.4
ANY STATE except Release Request, or Call Present State
Receive Unexpected RELEASE
Release Indication State 6.5.3.3
Inform PNNI call control to clear network connection, and after call control indicates that the VC is released, send RELEASE COMPLETE to calling user. Release call reference. Stop all timers. 6.5.3.3
2
ANY STATE except Call Present State
Receive Unexpected RELEASE COMPLETE 6.5.6.4
Release Virtual Channel (VC), clear connection, release call reference, stop all timers, inform PNNI call control. 6.5.6.4
Yes 1st T308 exp.
ANY STATE
Send RELEASE with call reference, Start T308, Disconnect Virtual Channel (VC-first time) 6.5.3.3
Release Request State 6.5.3.3
T308 exp?
No
Receive RELEASE COMPLETE, or RELEASE. Release VC, and call reference. Stop T308 6.5.3.3
2
Yes 2nd T308 exp.
Release call reference 6.5.3.3

Figure 35. PNNI – Call/Connection/Release at Preceding PNNI Interface


Call/Connection/Release at Succeeding ATM PNNI interface
Null State
Receive SETUP message
Call Initiated State
Available Traffic Parameters, QOS, and Bearer capabilities
Yes
Send CALL PROCEEDING message to proceding side. 6.5.2.4
Call Preceding Sent State (process call) 6.5.2.4
No
Send RELEASE COMPLETE to preceding side with crankback information. 6.5.2.3.2, 6.5.2.4, 6.5.3.2, 6.5.4 (call collision)
3
Receive Indication from Call Control that the call has been accepted. Send CONNECT to preceding side. 6.5.2.6
Active State
Receive indication that called party is ALERTING 6.5.2.5
Send ALERTING message to preceding side 6.5.2.5
Alerting Delivered State 6.5.2.5
ANY STATE except Release Request State
Receive Unexpected RELEASE
Release Indication State 6.5.3.3
Inform PNNI call control to clear network connection, and after call control indicates that the VC is released, send RELEASE COMPLETE to calling user. Release call reference. Stop all timers. 6.5.3.3
3
ANY STATE
Receive Unexpected RELEASE COMPLETE 6.5.6.4
Release Virtual Channel (VC), clear connection, release call reference, stop all timers, inform PNNI call control. 6.5.6.4
Yes 1st T308 exp.
ANY STATE
Send RELEASE with call reference, Start T308, Disconnect Virtual Channel (VC-first time) 6.5.3.3
Release Request State 6.5.3.3
T308 exp?
No
Receive RELEASE COMPLETE, or RELEASE. Release VC, and call reference. Stop T308 6.5.3.3
3
Yes 2nd T308 exp.
Release call reference 6.5.3.3

Restart is the same as UNI 4.0, but can be initiated by either side of a PNNI interface in order to return to an idle condition. Restart collisions are handled independently to avoid problems. 6.5.5
Clearing is normally initiated by the UNI Call Clearing Request, but any failure, Administrative Action, or exception may allow the preceding or succeeding side to initiate clearing. 6.5.3.3

Figure 36. PNNI – Call/Connection/Release at Succeeding PNNI Interface

Each node participating in routing keeps a copy of the original SETUP message, until a CONNECT or ALERTING message is received from the following node. Returned Crankback messages contain information pertaining to a blocked node/link. Having a copy of the SETUP message facilitates the possibility of a SETUP message reroute with an alternate DTL.

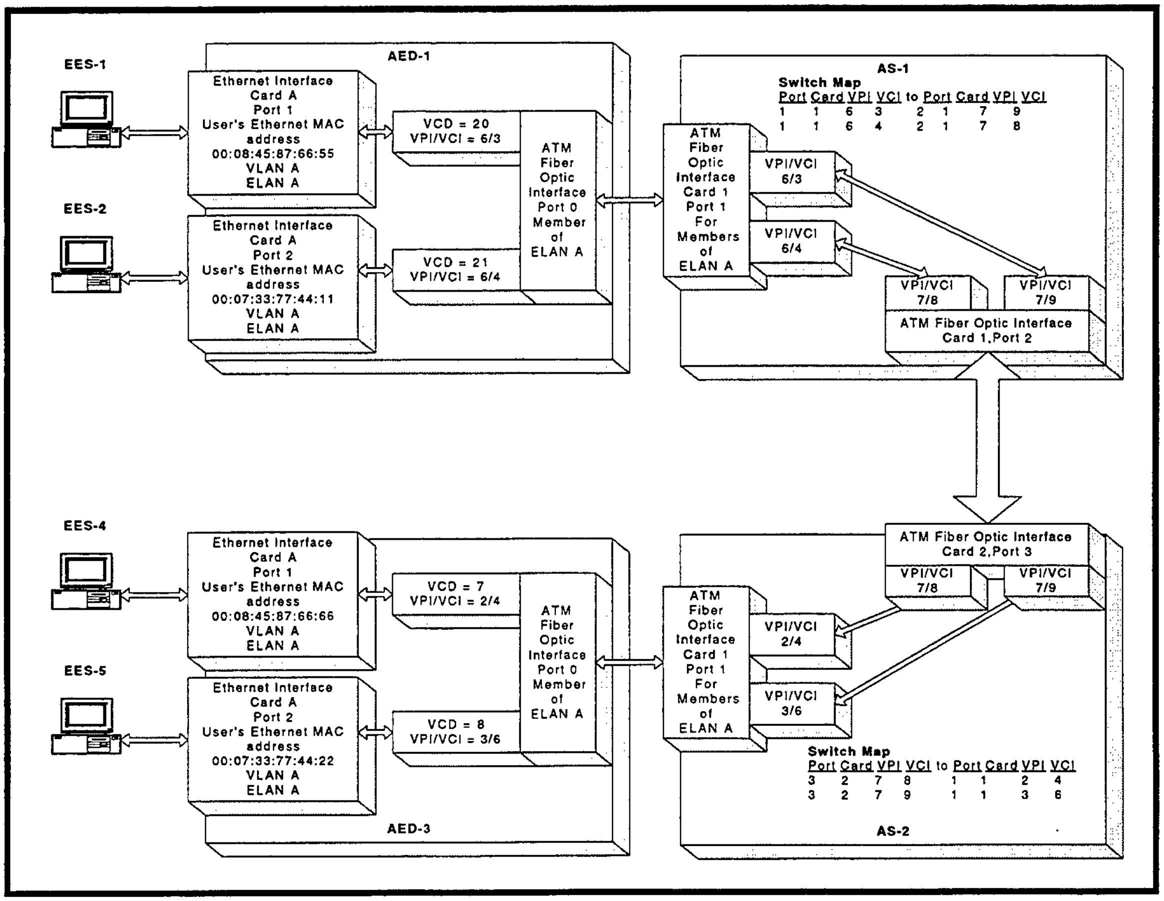
G. MANAGEMENT PLANE

The Management Plane is a distributed set of protocol management functions, defined within the protocols themselves. Management Plane functions are described within the various protocol sections of this document.

V. DATA TRANSPORT EXAMPLE OVER NETX

As previously stated, all remote Ethernet networks on NetX are interconnected by an ATM network. The ATM network is composed of ATM Switches and ATM edge devices. The ATM Edge Devices specific to NetX are either a Router, or Ethernet switches attached to the ATM network with an ATM interface card. The purpose of a Edge Device is to provide a gateway for dissimilar network connections (Ethernet and ATM). An Ethernet network transmits in units "frames" no greater than 1518 bytes; an ATM network transmits in units "cells" of only 53 bytes. Addressing is different. IP is connectionless; ATM is connection-oriented. A host of other differences exist. The NetX ATM backbone uses LANE to emulate an Ethernet network so that its operation is transparent to an end user (EES).

What remains to be explained, with regards to transmission of NetX user data is the logical flow of information at the Edge Devices and ATM switches, and an overview of addressing that is used on an established communications path between two EESs.



EES-1
EES-2
EES-4
EES-5
AED-1
AED-3
AS-1
AS-2
Ethernet Interface Card A Port 1 User's Ethernet MAC address 00:08:45:87:66:55 VLAN A ELAN A
Ethernet Interface Card A Port 2 User's Ethernet MAC address 00:07:33:77:44:11 VLAN A ELAN A
Ethernet Interface Card A Port 1 User's Ethernet MAC address 00:08:45:87:66:66 VLAN A ELAN A
Ethernet Interface Card A Port 2 User's Ethernet MAC address 00:07:33:77:44:22 VLAN A ELAN A
VCD = 20 VPI/VCI = 6/3
VCD = 21 VPI/VCI = 6/4
VCD = 7 VPI/VCI = 2/4
VCD = 8 VPI/VCI = 3/6
ATM Fiber Optic Interface Port 0 Member of ELAN A
ATM Fiber Optic Interface Card 1 Port 1 For Members of ELAN A
VPI/VCI 6/3
VPI/VCI 6/4
VPI/VCI 7/8
VPI/VCI 7/9
VPI/VCI 2/4
VPI/VCI 3/6
ATM Fiber Optic Interface Card 1,Port 2
ATM Fiber Optic Interface Card 2,Port 3
Switch Map
Port Card VPI VCI to Port Card VPI VCI
1 1 6 3 2 1 7 9
1 1 6 4 2 1 7 8
Switch Map
Port Card VPI VCI to Port Card VPI VCI
3 2 7 8 1 1 2 4
3 2 7 9 1 1 3 6

Figure 37. The Mapping of User Data from Source to Destination Ethernet Device

Each EES on NetX (See Figure 1) has four unique labels, an IP address, a MAC address, and two not previously mentioned; a port number and card number associated with the AED port to which it connects. (See

Figure 37.) In Figure 37 no two EESs share a port on an AED on NetX. Note that a AED port can connect to a shared Ethernet segment which can have several EESs; however, every EES connected must be a member of the same Virtual LAN (VLAN). The four labels noted (with respect to NetX) distinguish one EES from other EESs attached to an AED.

Each Edge Device is given a label to identify the group of users directly connected to it; this is called a Virtual LAN label, like "VLAN A" or "VLAN B," etc. Virtual LAN technology is not standardized. Cisco's VLAN implementation will permit only VLANs with the same VLAN labels to communicate. VLANs with dissimilar labels can only communicate if routed.

Cisco provides a means of mapping each Ethernet user to a VLAN and each VLAN to an ATM LANE label. ATM LANE labels are called ELAN labels; an example ELAN label would be "ELAN A."

ATM transmits information in small units called "cells.:" ATM cells are 53 bytes long and each contain a 5-byte cell header. An ATM cell header contains a VPI and a VCI identifier. These identifiers are assigned either permanently or on-demand for a circuit connection and identify each cell uniquely within each device as that for a particular established ATM user connection. A VPI/VCI identifier example may be "6/3."

When an AED receives an Ethernet frame from a local Ethernet port that is destined to be sent over the ATM network, LANE procedures within that edge device will assist other ATM protocols in establishing a circuit to the remote destination. After the ATM circuit is setup all ATM devices including the edge devices have VPI/VCI values established for this connection.

A Cisco Edge Device also assigns a unique Virtual Circuit Descriptor (VCD) to the Users' VLAN/ELAN connection for the purposes of mapping each VLAN circuit to the correct established ATM circuit. The remote edge device also does similar mapping.

VI. PROTOCOL SECURITY SUMMARY

Network security inherent in ATM and LANE protocols issued prior to 1999 are weak. An ATM Forum Security Specification published in February 1999, and follow-on UNI 4.0 and PNNI Security Addendum's published in May 1999, provide additional security measures; however, as of this writing they have not been incorporated into Fore Systems, nor Cisco Systems ATM products referenced in this document. This thesis does not reflect the use of the noted February and May security publications.

Vulnerabilities related to ATM and LANE reside in the fact that ATM attached devices are typically trusted. An ATM attached station with respect to LANE can be a LEC, LES, BUS, and/or SMS. LECs are provided with addressing and TLV information of other LECs and LANE services in their ELAN. The users of trusted devices generally have access to detailed topology and other network specific information that is not advertised outside a trusted environment.

High level topology and other network information is typically advertised in public web servers. Although public web provided information is not typically detailed it can provide enough information to narrow down possible network vulnerabilities that can be used to compromise a network.

If a network ATM/LANE attached client has office automation software such as email, web browsers, etc., it is at risk of being compromised. Even network management stations typically have internet access for patch downloads, investigating publicly available product information such as solutions to common problems. Access to the outside world in most cases is a necessity, but it means that every client on a trusted network should no longer be trusted. Hacking is all too common. If a system is hacked then detailed network related information is available outside the trusted domain. Network designers must re-evaluate network security with a mindset that their networks connect hosts that can not be trusted, and try to limit possible resulting security-related vulnerabilities to protect other hosts and the network as much as possible.

Trusted ATM/LANE attached edge devices learn about network topology and addressing via UNI ILMI and LANE LE_ARP cache entries. Compromised and trusted rouge Edge Devices have the ability to bypass routers and LANE client membership control. Changes to local and/or remote addressing related information and/or software facilitates the bypassing of router and LANE access control procedures. Address scanning can also be performed, as well as unauthorized in-band SNMP or telnet access to network devices.

Physical access to network cabling or electronic network equipment is dangerous from a security perspective. Network cabling (above and below ground) can be damaged or present a means to access network communications. Access to electronic network equipment such as routers, ATM switches, or ATM connected edge devices is dangerous because devices could be turned off, damaged, compromised through vendor specific emergency password recovery procedures, or accessed through console ports.

There exists the possibility of transmission errors either occurring between connecting network nodes, or within them. Various measures such as sequence numbers, transaction identifiers, checksums, and cyclic redundancy codes, along with program state checks are only a subset of the many mechanisms used by network devices and end user devices to insure that un-errored data is delivered to the proper destination. The culmination of these checks and processes noted within this thesis suggest that possible data corruptions occuring between Edge Devices are detected and if not corrected are not passed through Edge Devices to EESs. Adding TCP/IP related protocol checks and processes at Ethernet End Stations (EESs) into consideration make the possibility delivery of application frames to similar applications on wrong destination computers due to bit or burst errors extremely unlikely.

A. RESULTANT CAPABILITIES AND MEANS OF COMPROMISING A NETWORK OR HOST

This section suggests the following capabilities with respect to vendor-independent ATM and LANE protocols implemented without any vendor supplied security features or settings. The scope of this section is within the confines of an ATM backbone and does not consider VLAN or TCP/IP related protocols of EESs or AEDs.

1. Modifying Local Host Software to Spoof other Devices

A rogue LEC can spoof (pretend to be) other LANE LECs or LANE Services. An AED can modify its local ATM software such that it allows its primary ATM and/or MAC address to be changed without terminating its ELAN membership. An AED may also be able to modify its local software such that it can modify or add local ATM and/or MAC addresses without trying to create new ELAN memberships. Typically software changes are needed (this excludes most ATM test equipment) because LANE mandates ELAN termination with local primary address changes, and new ELAN joins (non-proxy LEC only) with new local addresses learned.

Note that the above software modification only applies to edge connected ATM workstations authorized to be LECs or LANE Services. Having an ATM switch, or a Catalyst 5000 switch with an ATM module can negate the need to modify device software.

Spoofing LANE devices may require the modification of LE ARP caches or cache software so that local LE ARP cache entries of unicast MAC/ATM address bindings of Target Hosts (TH)s on other ELANs can be entered. LANE only allows multicast or group addresses to be manually entered into the LE_ARP cache, but not unicast addresses. Spoofing may also require the installation of local Proxy LEC or PNNI software.

2. Router- ATM/MAC Binding Changes on TH's

A TH LEC could have its MAC-ATM address associated with a default router address binding changed in its LE_ARP cache. The infection could scan and replace memory, replace the software that is responsible for maintaining a LE_ARP table, or take place via other means discussed later. Although vendor software is typically delivered in compiled format, in at least two cases I have found source code for ATM equipment available on the internet.

If a Rouge Host (RH) is on the same ELAN as a TH, and masquerades as a router only to the TH, it is performing a "Man in the Middle" type of attack. The attack sequence is as follows:

1. A TH and a RH are LECs joined to the same ELAN.
2. The RH modifies a THs LE ARP cache with a virus or modifies it by some other means such that the MAC address associated with an Ethernet Router (Default Gateway) no longer is bound to the real router's ATM address. Instead, the TH's LE_ARP cache has the RHs ATM address bound to the real MAC address of the router.
3. The RH needs to have modified software on its system. The software change results with an acceptance of a data direct VCC SETUP requests from a TH for the routers MAC address. The RH can copy all packets from the TH. In this scenerio the RH must setup a VCC connection to the real router using the TH's ATM address and MAC address. This should not be a problem setting up because VCC SETUP requests are processed and routed according to the TARGET-ATM-ADDRESS within the SETUP message, not the SOURCE-ATM-ADDRESS. Once a bi-directional VCC is setup to the router, the router makes a connection with whomever the TH is wishing to communicate; data will just be returned along the VCCs setup. A RH can copy and rerouted data it receives from the router back to a TH.

3. LECs Changing LE ARP Cache Entries of other LECs

Proxy LECs are required not to register any remote MAC addresses or Route Descriptors that they represent in an ELAN. Proxy LECs are source route and/or transparent bridges that typically learn new LE ARP MAC/ATM address bindings from received packets. Non-static unicast MAC/ATM address bindings age out slowly. One problem with this is that because of the slow aging process, device connections that are moved from one Proxy LEC to another won't be represented correctly within the ELAN. Both the old and new Proxy LECs will respond to LE ARPs until the MAC/ATM address bindings on the old Proxy LEC age out. This problem is noted in the LANE [4] standard.

If an invalid Proxy LEC MAC/ATM address binding is associated with a bridge, BPDUs and LE_TOPOLOGY_REQUEST messages are sent to a BUS and a LES respectively, and forwarded to all LECs. These messages cause associated bridge address bindings to age quickly. A device must attempt to verify, or update aged address bindings before removing it from its LE ARP cache.

A "LE_NARP" message (called the "Targetless LE ARP in LANE v2) can be sent by LECs or Proxy LECs to other LECs and Proxy LECs informing them of changed address bindings.

A No-source LE_NARP_REQUEST can be used by a LEC to tell other LECs that it no longer represents a MAC/ATM address binding. The source of No-source LE_NARP_REQUEST does not include its source address within the message, it has to be set to "all zeros."

No validity checking mechanisms were found for incoming LE ARP associated messages at the LES except that they must have the same bindings (did not specified which bindings) and TLVs as the most recent

LE_REGISTER_REQUEST received by the LES. No mechanisms were found for an LES checking SOURCE_ATM_ADDRESS information on Control Direct VCCs.

The LE_TOPOLOGY_REQUEST, LE_NARP_REQUEST, Targetless LE_ARP_REQUEST, and No-source LE_NARP_REQUEST messages may be exploited by a RH to alter bindings of THs. The exploitation of these messages by an RH may be to cause Denial-of-Service attacks on THs or to redirect connection requests to a RH for unauthorized means such as that noted in Capability 2 (above).

4. Removing a LEC or Proxy LEC from an ELAN

A LEC typically sends a LE_UNREGISTER_REQUEST to a LES to remove MAC/ATM bindings from a LESs' registration database. Once a registration is removed all associated control and multicast VCCs to ELAN services are dropped. A LES only matches REQUESTOR_LECID and LAN_DESTINATION information on received LE_UNREGISTER_REQUESTs in order to validate that the source of the message was the source of the original registration message.

An RH may alter its software to change the REQUESTOR_LECID and LAN_DESTINATION information within a transmitted LE_UNREGISTER_REQUEST message. This exploitation could be a Denial-of-Service attack or an attack that would allow a RH to join an ELAN as a TH without being denied because of duplicate address registrations.

5. Registering as Another Client with an Alternate Multiplexing Scheme

It may be possible within LANE v2 for someone to register someone else's ATM address within LANE, but with a reversed multiplexing scheme. The implications of this, if proven, are unknown.

6. Re-registering a Client with changed Type-Length-Value (TLV) information

It is possible to re-register a MAC address - ATM address pair, with different TLVs. This results in the clearing, then replacement of all existing TLV settings with those received in the re-registration request. An RH may use this to cause a Denial-of-Service attack on a TH.

7. Registering with UNI as a LECS Using an Active LECS ATM Address, or an LECS "well-known" ATM Address

A LEC join process typically begins with a LEC trying to connect to a LECS. It first looks for a preconfigured LECS address and if found uses this address in an UNI setup message to connect to it. If it has no local preconfigured LECS address then it uses ILMI to discover an address of a LECS. If this discovery fails then it uses the ATM Forum "well-known" ATM address.

There are no statements in LANE or UNI to forbid devices from registering (with UNI) a local address as a LECS address, an already registered active LECS ATM address, or with an ATM Forum defined LECS "well-known" address (resulting with multiple "well-known" address registrations in a network).

It is possible that a RH can register as a LECS and accept calls from THs. If an RH also has LES/BUS services configured then it can let THs join spoofed ELANs.

The exploitation of this vulnerability can enable an RH to learn ELAN names, ATM addresses, or TLV bindings associated with LECs, or even LANE services trying to join an ELAN. If a LES spoofing RH can configure LECs on its workstation when LE_ARP_REQUESTS arrive, then (with a little homework) it may be able to spoof another host on an ELAN and learn login user names and passwords. If LNNI v2 "Synchronization Peer Servers" are setup between valid LESs and SMSs, and redundant LECS are not enabled or configured then the scope of learned information is limited to information gained by clients requesting configuration. The scope of information learned can be greatly increased if Capability 8 (below) were exploited.

8. Spoofing LECS, LESs, BUSs, when LANE v2 Services are Configured with Multiple LECS and "SynchronizingPeerServer" Lists

If a LANE v2 domain contains more than one LECS, and ILMI is used to discover other LECS, or if redundant LECS ATM addresses are preconfigured on valid LECS then a RH may be able to compromise an entire network.

If an RH spoofed a LECS according to Capability 7 (above), and set up new LES/BUS services after LECS synchronization via LNNI v2 LECSSYN VCCs. It is possible to infect and update valid LES and SMS SynchronizationPeerServer lists so that rouge LESs and SMSs can gain all active client and server registration information within an entire LANE domain. Only the SOURCE-ATM-ADDRESS of a LECS is checked by other LECS to authenticate LECSSYN VCC establishment. LES and SMS SynchronizationPeerServer lists can be updated by a LECS via CONTROL DIRECT VCCs. The result of this type of attack can compromise every host connected to and within an ATM LANE network.

9. Setting up a Cache Synchronization (CACHESYN) VCC and Using Server Cache Synchronization Protocol (SCSP) to Become a "SynchronizingPeerServer" to a LES (Applies to a SMS as Well)

A LES must accept UNI signaling messages to accept CACHESYN VCC setup requests from an address it already has a connection to. This VCC may be released only if the following two conditions are met: (1) no valid synchronization or control communications use the VCC within a uses defined time, AND (2) the calling parties ATM address is not on its SynchronizationPeerServer list [5].

If an RH sends a UNI LNNI SETUP message to a LES for a CACHESYN VCC it must accept it. After a CACHESYNC VCC is setup the RH begins the SCSP by sending "Hello" messages.

Note that a RH must know what the MAC address is of a Target LES or SMS. Knowing the manufacturer of the connecting device could reduce the number of possible MAC addresses to around 16, 777 which could be incrementally tried in successive SCSP Hello messages very quickly.

When a LES or SMS gets a SCSP Hello via its CACHESYNC VCC that has a Receiving ID that matches its own, it stores the Sender ID of the received Hello message and updates its peer list with the senders ID. If an interpretation of the SCSP and LNNI standards is that the "peer list" and the "synchronization peer server" list are the same list then the RH is added as a new LES to the THs "SynchronizationPeerServer" list. Registration information about every device active on the ELAN is then disclosed.

The SCSP protocol supports an authentication extension, however no suggested authentication methods were suggested for its use. Note the following paragraph taken from the SCSP specification.

> If authentication extension is not used, or if the security is compromised, then SCSP servers are liable to both spoofing attacks, active attacks, and passive attacks. ... Any SCSP server is susceptible to Denial of Service (DOS) attacks. A rouge host can inundate its neighboring SCSP server with SCSP packets. ... If security of any SCSP server is compromised, the entire database becomes vulnerable to corruption originating from the compromised server. [6]

If the interpretation of the LNNI v2 specification and SCSP protocol as noted above is correct and a RH exploits this vulnerability then all active hosts that are members of the ELAN associated with the TH's LES are subject to all types of attacks.

10. Setting up a Multicast Forward VCC to 802.1D LECs

A LANE v2 LEC must accept all Multicast Forward VCCs. A LEC can optionally use a LE_VERIFY procedure to check the ATM address of the Multicast Forward VCC source to insure that it is a BUS. An 802.1D LEC (Proxy LEC) receives configure BPDU packets from a BUS. If an RH spoofs a BUS for the sole purpose of sending poisoned configure BPSU packets then bridge loops may result causing Denial-of-Service attacks.

11. Directly Connecting to LECs in other ELANs within a LANE Domain

The UNI registration process involves registering any LANE addresses associated with LEC variables C6 (local uni-cast address list), C8 (local route descriptors list), and C15 (multicast MAC address list). If addressing information associated with these variables were changed and LANE connection management procedures were bypassed then it would be possible to setup connections to LECs or LANE services in other ELANS without ever registering with the THs ELAN. This is possible because after an ATM attached station completes the UNI ILMI address registration process, ATM circuits are setup using only "DESTINATION-ATM-ADDRESS" field information in the UNI setup request. The UNI setup request is over a pre-established signaling channel and the call setups create bi-directional Data Direct VCCs from the caller to the calling party. With LANE v1 only non-multiplexed VCCs are used, and ELAN-ID fields are not defined. The TH checks the Called party ATM and MAC address information, and may check the Calling party ATM and or MAC address information before accepting the connection. Note that both the calling ATM and MAC address information can be changed to whatever the RH desires.

A TH LEC must check (though an un-mandated means) LE_ARP cache information before "idle timeout" associated address deletions. One of the suggested ways in the LUNI v2 standard is that a LEC can check aging addresses by looking at unicast MAC addresses on incoming VCCs. If this mechanism is used to validate addresses associated with data direct VCCs then anyone having a data direct VCC to a LEC can validate themselves to that LEC. An RH exploiting this vulnerability in combination with possible gained capabilities associated with possibly compromising trusted hosts may enable any type attack on a TH or other hosts on the network.

12. Iterative Testing of Unused Message Fields and Vendor Specific Protocol Extensions

Undocumented vendor specific TLVs extensions may be implemented in LE_CONFIGURE_REQUESTs and responses. Vendor specific extensions are also provided by SCSP extensions. Undocumented extensions may provide back door capabilities that are only known by a vendor. Only iterative testing of unused fields and vendor specific extensions could reveal the presence of these types of vulnerabilities.

13. Monitoring ATM Traffic on Fiber Optic Cables

ATM test and monitoring equipment can connect to passive fiber optic splitters and monitor traffic. Many types of splitters may be used, some require cable cutting and splicing, others may take advantage of fiber microbending effects and not even require a fiber to be cut. ATM test equipment such as a Radcom Inc., PrismLite automatically descrambles SONET/SDH and ATM cell payloads when connected to passive fiber optic splitters providing ATM traffic.

14. Damaging Communication Cables

Any exposed or buried fiber optic communication path could be physically damaged resulting in a Denial-of-Service attack. The consequences of this type of action are reduced if redundant LANE services are offered via LANE v2, and/or redundant fiber optic paths are incorporated into the network cable plant design. Note that LANE v1 does not offer redundant LANE services.

15. Registering ATM Addresses that are Already Being Used in an ATM Domain

UNI does not require that ATM addresses registered via ILMI be unique. Multiple devices within an ATM network can register the same ATM addresses. Registered addresses are PNNI propagated either completely, or in a summarized form. PNNI will route calls to destinations that have the longest Most Significant Address Bit (MSAB) matches to entries in local routing tables.

16. Joining an ATM Network as a PNNI Node

Any device connecting to an ATM switch can join as a PNNI node. PNNI has no authentication, and trusts devices connecting to it. ILMI network prefixes, LE_ARP cache information, and a general knowledge of a network naming convention and topology would suffice in attempting to connect to a switch via ILMI and registering as a PNNI node. A RH that has registed itself as a PNNI node on an operational is capable of viewing all routing information via newly formed RCCs, infecting routing tables with poisoned connectivity information, and can even limit address learning from or to upper level hierarchical peer groups.

B. TCP/IP PROTOCOL SUITE SECURITY SUMMARY

Every EES in

Figure 1 has TCP/IP software installed. This software adds further protection against wrongful delivery of application PDUs due to errored bit or burst errors along a communication path.

As shown in Table 2, there are at least 4 layers with between 19 and 23 separate fields of encapsulated information involved with the receipt, and delivery of data to a user application. In terms of addressing and ports, the frame would need to have an incorrect 48-bit MAC address, 32-bit IP address, with the correct 16-bit source port number, and 16-bit destination port number to be delivered to the wrong destination and received by the correct application.

There are between two and three checks to validate the integrity of the data; the 32-bit CRC check on the received 802.3 frame, a 16 bit checksum on the IP packet header, and a 16 bit checksum at the TCP layer, or (optionally) the UDP layer. Note the TCP and UDP checksums are calculated not only over their associated headers and data, but also over the IP layer information starting with the source IP address.

There are between two and three length fields, which if incorrect can lead to frame information being discarded. They are the 802.3 frame length field (which includes all data up to but not including the CRC field), the IP Datagram length field, and the TCP or UDP length fields. Note if any length field is incorrect, due to an error in transmission, it would most likely be detected by the CRC and checksum processes.

The TCP protocol provides a means to deliver data (segments) in order. Duplicated segments received from lower layers are discarded, but if an error produces an incorrect (future sequence number) the correct segment may be discarded. If information is given to an application incorrectly, the results are unknown and application dependent. An application may check all the information it receives from the TCP layer before it processes it.

VII. PROTOCOL/PLANE SPECIFIC SECURITY POINTS AND DISCUSSIONS

Chapters VI and VI.B provide an overview of some suggested capabilities and qualitative security related remarks. The following sections offer some protocol and/or plane specific security points and discussions. Recall, "planes" are a made up of a collection of protocols. Note that because this Chapter references "planes", some protocol related information is restated, however it is not a complete listing of security related information.

A. GENERAL POSITIVE SECURITY POINTS ABOUT LANE

LANE has some mechanisms built-in to control ELAN membership, and the validity of received messages. Some of these mechanisms, which are protocol specific, are given below.

1. If any of the control VCCs between the LEC and LES fail then the LECs ELAN membership is terminated along with any associated control VCCs.
2. Restrictions are in place to prevent duplicate MAC address registrations, and ATM address registrations with the same multiplexing scheme on an ELAN. Note that this capability is specific to a fully compliant LNNI v2 implementation.
3. The SCSP optionally provides a message authentication extension. Note, however that it suggests no method to utilize this feature.
4. The LE Header Source ELANID is required and checked at destination hosts for the successful delivery of information for LANE v2 LLC-multiplexed data frames (this is not an option with LANE v1 where the LECID is only checked by the sender to filter out its own data frames).
5. A SCSP packet contains a 16 bit Checksum which is calculated over the entire SCSP packet including link layer and/or other protocol encapsulation. Packet size will vary, as it is dependent upon the message type (CA, CSU, CSUS, or Hello), as well as the content included within each message.

B. THE SONET/SDH PHYSICAL LAYER

Security related vulnerabilities asscociated with errors at the SONET/SDH physical layer are unlikely. The physical layer has built in mechanisms to detect LOS, LOF, LCD, errored HEC, as well as mechanisms to monitor and report errors, and the extent of errors to local management, and to remote devices. All monitoring, performance, and detected problems are transmitted between devices within the SONET overhead. UNI excludes the use of all SONET/SDH fields not specified in the UNI 3.1 standard for use. It does this by setting all non-UNI specified fields within the SONET/SDH field to "1" at the physical layer before the frame is scrambled and sent out on the transmission medium. This means that SDH specific services such as "order-wire" and other control functions in the SDH standard cannot be implemented.

ATM test equipment is capable of unscrambling and viewing each SONET/SDH frame and ATM cell. At the SONET/SDH level all addressing is in terms of paths and channels, not ATM addresses. One would expect that if specific ATM addresses are targeted for monitoring etc, that signaling channel information would be collected too (after unscrambling ATM cell payloads), to discern associated VPI/VCI pairs with the target.

If someone were to cut a fiber, within 100 microseconds a Loss of Signal (LOS) would be detected by each interface associated with the cut. An alternative to cutting fibers is to use a device that uses micro-bending to redirect a small portion of light from a fiber optic cable for monitoring traffic. This information could be processed later so that an ATM device could be configured to act on the behalf of each device that it is between. When a disruption in power occurs, possibly purposely done during a lightening storm, a fiber optic cable pair could be cut and a preconfigured RH could be inserted between the two ATM devices it is spoofing.

The latency through a switch on a pre-established circuit is on the order of 10 microseconds. Buffering mechanisms implemented in ATM switches that compensate for Cell Delay Variations should negate any monitoring attempts to detect delay deviations that exceed this value.

In summary, could a user accidentally or intentionally send data over a node that, after being scrambled at the cell payload and SONET/SDH frame level, create a loss of synchronization? An analysis performed by Lucent Technologies states:

> In order for a malicious user to successfully cause all zeros or all ones, the user has to successfully guess the state of the ATM scrambler which can be one of 2^{43} sequence an therefore this probability is negligible. (personal note – the SONET/SDH frame scrambler would then scramble this information). The user can generate input data sequences which will cause 43 bit periodic patterns on the line. ... as long as synchronous transmission equipment can withstand small numbers of transitions (such as 6, 8, or 10) in 43 bit periodic patterns with the pattern repeating up to 46 times in an OC-3, any malicious attack will not be successful. ... any bit error in the physical layer produces two bit errors (43 bits apart) after being descrambled.[10]

Additionally, noted below are ANSI comments which suggest that the use of ATM HEC correction is not recommended:

> When in the Correction State, the HEC error detection capability is reduced. It has been shown that when the header is corrupted with 3 or more bit errors, for some patterns the HEC fails to detect multiple bit errors, the HEC treats such headers as containing only a single bit error and attempts to correct them. This usually results in a non-discarded cell with bit errors in the header.[11]

The use of HEC is an open issue. It is beneficial to use it in certain environments, especially those that produce single bit errors (possibly fiber optic communications lines); however, the above negative HEC correction impact should be taken into consideration. The use, or non-use of HEC correction was not investigated further within this thesis. Recall, that addressing and network size play important roles in the prevention of security related vulnerabilities associated with bit or burst errors.

C. SECURITY POINTS OF THE ATM LAYER

One way to detect the insertion of a device between two nodes connected along a known path is to measure and record the Cell Delay along that path and compare it to delay measurements made to new connections made along that same path. There is no requirement, however, for the ATM layer OAM Loopback cell capability to support cell delay measurement. Along the lines of OAM, in terms of detecting errors, an OAM cell has a CRC-10 field provide a means to detect errors within an ATM layer OAM cell. A Cell Delay Variance (CDV) Tolerance is a measure of a delay variance due to the multiplexing of cells from two or more ATM connections. The CDV should not be used for insertion detection because a clumping effect of cells is expected at Public UNI interfaces or procuced on Private UNI connected end systems [7].

Appendix A.6.2.3 of the UNI [7] notes that the CMR is primarily influenced by undetected/miscorrected errors in the cell header, which is in turn primarily influenced by the transmission error rate. The likelihood that an errored cell passing through the physical layer to the ATM layer maps to a valid VPI/VCI is dependent upon the number of assigned VPI/VCIs, and those VPI/VCIs actively being used at the time of the error. The smaller the network the less likely that this can happen.

A suggested means for measuring Cell Misinsertion Rate (CMR) is provided by ITU-T [12]. Basically a dedicated VP or VC is created between two nodes that do not transmit cells. Any ATM layer cells (excluding OAM cells) received by either node are miss-inserted cells

1. Positive Security ATM Address Point

In terms of security, having 20 Bytes (160 bits) associated with an ATM address allows 10^48 possible addresses. Having this many addresses means that probability of an ATM address having a random bit or burst error that augments an ATM address such that it pertains to a valid address within locally administered and controlled network address domain is unlikely.

2. Positive Security Related UNI Address Registration Points

1. UNI addressing related information is deleted when an interface is initialized at start-up.
2. UNI performs some information validity checking at both the user-side and network-side when addresses are being registered.
3. Addressing related information associated with a failed UNI link are deleted.
4. The UNI standard allows networks to reject unauthorized addresses based upon user defined rules.
5. UNI prevents the registration of duplicate full ATM addresses, however the scope of this restriction is not defined and may be implemented on a per port basis.

3. Negative/Uncertain Security Related UNI Address Registration Issues:

1. UNI Address Registration allows a user to supply its own Network Prefix. If this is not restricted then the geographical scope of address spoofing is increased.
2. Address validity checks at the network-side and user-side are not fully defined.

3. It takes (by default) 20 seconds for a UNI to realize a link has failed. (this however, can be changed).
4. UNI specifies no required action for duplicate address registrations.
5. No information was found with respect to UNI de-registering users sitting behind an ATM Edge Device, other than that the user-side of the UNI interface is responsible for de-registering an address.
6. Registration checks are not fully defined in the UNI standard.
7. Fore only checks duplicate ATM addresses per port , this allows duplicate ATM addresses to be registered on a switch (different ports), or on a Network.
8. ATM address scan attacks, or educated address attacks on networks is rather simple. Out of a 20 Byte ATM address the first 13 bytes are provided by a switch, having LE_ARP cache listings provide other switch prefixes as well as the manufacturers of ATM adapter cards which predefine the most significant 3 bytes of End System Identifiers. Default Selector bytes use by manufacturers, or combinations of sample selector bytes in LE ARP caches may offer clues. Unless a well thought out random ATM addressing scheme is implemented guessing ATM addresses should be a relatively simple task.

D. SECURITY RELATED SUMMARY OF ATM USER PLANE

It is unlikely that bit or burst errors along a data communications path will result in incorrect destinations receiving and processing corrupted data frames.

All Ethernet packets that are passed across an ATM backbone are checked at the receiving end with a 32-bit CRC in the AAL-5 CPCS. Other fields in the AAL-5 CPCS such as Length, PAD, and possibly the CPCS-UU play a role in the successful delivery of the CPCS payload data to the LEC layer.

The ATM layer checks VPI, VCI, and PT values for accuracy. Only specific header values associated with pre-established connections, or pre-assigned reserved values, can pass above the ATM Layer; the rest are discarded. There are 27 header bits associated with any Edge device VPI, VCI, and PT, and 31 header bits associated with VPI, VCI, and PT fields between ATM switches. Using the maximum allowable bits in these fields results in 134 million possible associated VPI/VCI/PT combinations at an edge device, and 2.1 billion possible combinations between ATM switches. The possibility of an error resulting in match to an established connection, or a reserved VPI/VCI/PT combination is very small. Even if an errored cell is passed through the ATM backbone undetected, the Edge Device's AAL-5 CPCS performs a 32-bit CRC on its payload when it is delivered.

The Physical Layer performs an HEC check on the ATM cell header before it is allowed to pass to the ATM Layer. This HEC check can correct 100% single bit errors, and detect 84% of multiple bit errors [8]. When an error is detected and it is not correctable, or if the interface is configured to not correct errored headers, then the ATM layer is notified and the cell is discarded. All information passed between ATM entities such as ATM switches or Edge Devices is exchanged using the SONET/SDH standard. This standard defines a frame cell structure and procedures to insure data integrity. The OC-3 STS-3c frame and associated processes includes overhead for three Byte Interleaved Parity (BIP) checks, alarm and error reporting to upstream neighbors, and a ones-density check to monitor the performance of a transmission to insure correct delivery of information. Every ATM cell payload transmitted across the ATM network is scrambled, as well as the entire SONET payload.

A configuration note for Fore systems ASX switches needs to be taken into account when considering the security related functions offered by the above noted protocols. Fore systems ASX switches default to turn off Alarm Indication Signal(AIS)/Remote Defect Indication (RDI) OAM cell generation. Note that if it were turned on, it is only defined on a per port basis. SVCs and SPANS SPVCs do not generate AIS cells. OAM cells are generated only for through paths, originating paths, PVCs, and PNNI SPVCs that originate on the port where this function is enabled.

E. SECURITY RELATED SUMMARY OF ATM CONTROL PLANE

The control plane offers security rich features such as assured mode signaling, message validity checks, and control over connection identifier assignments. Despite these positive features the control plane's weaknesses overshadow it's many positive security points. Security vulnerabilities center around the fact that the control plane (UNI) does not check the validity of who places a connection request, nor checks to insure that duplicate addresses are not registered throughout a network or switch. UNI also enables the automation of break-in attempts by providing information about the cause for connection failures.

1. Positive Security Related Points to SSCF, SSCOP, and CPCS Operation with Respect to Signaling

1. Received length, length mismatch, buffer overflow, and CPCS CRC errors, as well as incorrectly formatted CPCS-PDUs are detected.
2. The CPCS delivers error information to the SSCOP for impact determination.
3. Congestion information is passed to higher layers, and peer for transmission speed adjustments.
4. Assured mode connections are setup between peer SSCOPs, and connection status is monitored.
5. Program procedures associated with SSCOP are monitored for errors, and resynchronization, and restart procedures defined for possible resolution.
6. State variables and sequence numbers associated with SSCOP PDUs, as well as timers kept by both transmitter and receiver insure correct, ordered delivery of SSCOP-SDUs .

2. Negative Security Related Point of SSCF, SSCOP, and CPCS Operation.

Invalid SSCOP PDUs (see definition above), received with error notification via the "corrupted data deliver" option by the CPCS are discarded without notification to the sender.

3. Positive Security Points Related to UNI Call/Connection Setup, Release, and Restart

1. The network-side of a UNI interface provides VPI/VCI connection identifiers.

2. Timers are implemented on both the network-side, and user-side to release calls in the event of connection creation problems.
3. If a device fails to send appropriate messages, releasing the call during setup, other devices will carry out those functions.
4. Q.2931 checks for accuracy or validity of the Protocol Discriminator, message length, call-reference number, procedural errors, message type errors, message sequence errors, Message Information Element (MIE) coding errors, MIE duplication, and MIE content.
5. Some measures of address and compatibility checking in addition to (4) above are done by Q.2931, however those measures are implementation dependent, and unknown at this time.
6. Release collisions have been carefully thought out, and should not a problem.
7. Signaling information is sent over SAAL Service Specific sub-layers to provide an assured mode connection.
8. CPCS does 32 bit CRC checking. CPCS header fields are also checked for accuracy.

4. Negative/Uncertain Security Points Related to Call/Connection Setup, Release, and Restart

1. The "STATUS" and "STATUS COMPLETE" messages may provide sensitive addressing or configuration information to a RH performing scan types of attacks.
2. After a UNI connection is established, no information concerning the release of this connection, if inactive for some time, was found. LANE manages non-multiplexed VCCs, and LLC-multiplexed data flows, not individual LLC-multiplexed connections within flows. Individual LLC-multiplexed connections within flows are managed (timeouts, etc) by the applications that created them.
3. UNI Address checking methods are in UNI 3.1 Section 5.8.5.1. They state that they check to make sure that the ATM address being registered has not already been registered. A call to ATM engineers (who wish to remain anonymous) at an ATM switch manufacturer indicated that their interpretation of UNI 3.1 Section 5.8.5.1 was that duplicate addresses were only on a port by port basis. This means that if someone tried to register an address a second time over the same connection, their ATM equipment will respond with an error, and deny the registration. But, if someone tried to register that same address on the same switch, but to a different physical port then he or she would be able to register it. The scope of this problem reaches beyond a single switch to all parts of an ATM network. A network either summarizes advertised addresses, or advertises them in whole if they are foreign addresses that cannot be summarized. No duplicate address checking is done. ATM PNNI network routing is based on the longest matching address it has in its routing table, or an administratively shortest path. The UNI address registration process is completely separate and not linked with the LANE address registrations. This duplicate address registration vulnerability makes it easy for someone to

spoof network devices or other hosts, especially if a RH changes non-primary ATM addresses to spoof other devices.

4. How easy is it to find someone's ATM and MAC address? Consider this: An ATM Prefix (first 13 bytes) of the PNNI peer groups ATM address is supplied by the Network side of a UNI, during an automatic ILMI registration process. The next six bytes of an ATM address are the MAC address bytes. The first three bytes of the MAC address are the Organizational Unique Identifier of the manufacturer of the adapter card which are publicly made available. So only three bytes of MAC address information and a selector byte are uncertain. LANE uses the selector byte to distinguish between ATM devices, but UNI does not. Vendors might have default selector byte values built into their configuration code to make the card installation less complicated. By guessing the OUI correctly, and doing some homework on that vendor's default selector byte settings, only 24 unknown bits of ATM address information of a potential victim may remain undetermined. This works out to about 16.8 thousand addresses to scan. Having a computer initiate the creation of multiple connections (and release those that are actually setup) simultaneously may result in a very short time frame to scan all 16.8 K addresses.
5. Often when a call setup fails to an active ATM address, the refusing end device typically returns a reason for the failure. This might allow a user to automate a setup program to retry setups and intelligently alternate field bit patterns to gain access.
6. The ILMI process operates over an unassured mode connection. This means that ILMI does not have the added error detection capability offered by SSCOP. Even without SSCOP, undetected errors are still unlikely due to error and process checks of AAL5, and SONET/SDH layers associated with the transmission.
7. After an ATM end station connects to a switch via the UNI signaling process, future call SETUP messages to ATM attached devices do not have to include the Calling Party's ATM address. ATM switches do not check the calling parties' ATM address unless they are explicitly configured to do so. This allows a RH to then register different secondary addresses with LANE. A RH is not able to receive any call SETUP messages associated with LANE registered addresses not registered with UNI. A RH is able to initiate call SETUP messages "pretending to be another user" because ATM connections are setup with VPCI values from source to destination by using the called party ATM address.

F. SECURITY RELATED SUMMARY OF PNNI SIGNALING

Like UNI, PNNI signaling has inherent security rich features such as those offered by SAAL assured mode signaling, message composition checks, etc., which enable it to detect and handle errors. PNNI provides a means to specify error handling procedures in the event of unrecognized or corrupt messages, or IEs. PNNI routing provides means to check the composition of routing messages, along with CRC's and Checksum's. Despite the positive security related aspects of PNNI signaling and routing it is not inherently secure. PNNI signaling messages allow default error handling procedures (that may be implemented for security reasons) to be overridden. A RH host, using learned information via UNI signaling and ILMI registration, can register as a PNNI node via ILMI signaling at startup and take control of a network easily.

1. Positive Security Points Related to PNNI Signaling

1. The PNNI signaling layer is above the SAAL, ATM, and Physical layers. These lower layers provide added error detection, and filtering capabilities (already discussed). The SAAL assured mode service is used by PNNI as an added mechanism to detect missing or corrupt packets, as well as link disconnections.
2. All PNNI signaling message, like UNI signaling messages, have five common Parts (See Figure 33) which have special emphasis with regard to PNNI error detection: (1) Protocol Discriminator, (2) Call Reference, (3) Message Type, and (4) Message Length, (5) General Information Elements. All messages with an incorrect Protocol Discriminator are ignored by PNNI. Errors associated with Call Reference result in either a call being cleared, or the message ignored. The clearing of calls accompanies the return of cause information. If a message type is not recognized, it is ignored. Unexpected messages, except RELEASE and RELEASE COMPLETE, are ignored and a STATUS with cause is returned. The Message Length has only a minor contribution, in that if the message length field length (not its value) is not the proper size the message is ignored. The General Information Elements are checked for validity with respect to Call State, sequence, and content. Messages with mandatory IEs missing are either ignored or result in the clearing of the call. Note that Action indicator and IE Instruction fields settings can override default error-handling procedures.
3. PNNI signaling timers assist with error detection, and are used within state machine programs to insure that calls/connections are cleared, or reestablished in the event of an error.

2. Negative Security points related to PNNI signaling:

1. Message Action Indicator bits (MAIBs) in Part 3 of a Signaling message can be used to override default error handling procedures within switches for unrecognized messages. MAIBs enable someone to learn the reasons for failed signaling messages, or allow PNNI messages to be passed though PNNI interfaces without message validity checking.
2. Message IE Instruction Field settings make it possible to learn causes of failures associated with IEs. IEs have network specific information.
3. The handling of STATUS errors, other than what was already noted are implementation specific.
4. The Message Length field value is not checked for accuracy. The handling of message length inconsistencies is poor. Nothing was found with respect to PNNI call control notification in the event of a Message Length error.

3. Positive Security points related to PNNI routing:

1. RCCs use the CPCS error detection capability (CRC-32, etc.) with the CPCS "corrupted data delivery" option to help it detect errors in the AAL-SDU.
2. PTSEs contain Checksums that are calculated over the logical node-ID and PGID from the PTSP packet, as well as the entire contents of the PTSE, except the Remaining Lifetime field to help it detect errors in the PTSEs.
3. If the Packet Length field in the PNNI header specifies a length greater than that received the packet is discarded.
4. If the Packet Type is not recognized in the PNNI header the packet is discarded.
5. If the received Packet Version is not supported or is different from what is expected it will be discarded.

4. Negative Security points related to PNNI routing:

1. The handling of parsing errors in the PNNI header type or length field is at the discretion of the implementers.
2. The PNNI Hello process initially relinquishes a nodes PGID, node-ID, AESA to its neighbor. This information can be used by a compromised node or malicious neighbor to gain access and control of the network. An example could be an ATM connected host that has LANE running. If a RH were to reconfigure their workstation to be a PNNI node, rather than a UNI connected host, their workstation would register via ILMI as a PNNI node. It would then start a Database Synchronization process that would result in the outside host having detailed PG routing related information.
3. No authentication process is defined.

G. SECURITY RELATED SUMMARY OF ATM MANAGEMENT PLANE

The ATM management plane is built into each ATM related protocol. Management specific security points are not separately distinguished from security points relating to the protocols themselves. Because ILMI operates over AAL5 and SNMP some high-level security points relating to the SNMP standard and to the AAL5 layer are noted.

1. ILMI over SNMP v1

All ILMI messages are formatted as specified in RFC 1157 (SNMP version 1), but without UDP and IP addressing. Each ILMI message, sent or received by an UME occurs over AAL-5.

a. Positive Security Related RFC 1157 Points

1. SNMP messages with version number mismatches are discarded.
2. ILMI SNMP messages with a Community Name other than "x494C4D49" are discarded.
3. SNMP messages with Abstract Syntax Notation One (ASN.1) parse failures are discarded.

4. An authentication scheme is used.
5. SMNP Request IDs are sent and checked on every response.

b. Negative/Uncertain Security Related RFC 1157 Issues

1. No information was found as to the extent of the size of the Request ID.
2. No information was found within RFC 1157 concerning the authentication procedure.

2. Positive Security Related AAL-5 Points

1. A 32-bit CRC is provided to check every CPCS payload,
2. Other fields in the CPCS header (PAD, CPSC-UU, CPI, Length) are involved in the correct transfer of information.
3. ATM and Physical layer checks as discussed in the U-Plane are also implemented.

VIII. FORE AND CISCO CONFIGURATION OPTIONS AND APPLICABILITY TO NETX SECURITY

Fore Systems, Inc., and Cisco Systems, Inc., are major ATM and LAN device manufacturers in the networking industry. In an effort to make this document applicable to a majority of the ATM and LANE customer base, Fore Systems ATM Switches (ASX models) and Cisco Systems Edge Devices (Catalyst models) are the components of NetX's makeup.

The analysis thus far considered possible security vulnerabilities and security points associated with my interpretation of the network standards. Each manufacturer may interpret networking standards differently, that is why ATM Forum Protocol Implementation Conformance Statement (PICS) documents were created and interoperability test equipment is made. Some companies have made a living selling standards based PICS compliance test equipment; these devices help resolve most interoperability problems during product development and testing. Interoperability may be achieved; however, each vendor has its own implementation, often with non-standardized added features.

To protect investments in product development vendors do not typically release device source code; they release precompiled programs. One must trust that the vendor has not written back doors into their software and that their implementation of standardized protocols and non-standardized added features do not have detrimental security related "bugs" or oversights. Note that some ATM source code is available on the internet.

The following is not a complete summary of vendor device features or configuration options, it is an overview of vendor specific implementation issues and security related suggestions developed from the analysis of this thesis.

A. FORE SYSTEMS ATM SWITCHES, WORKSTATION ADAPTERS, AND LANE

The following reflects comments to Fore Systems ATM Switch Network Configuration Manual, Revision A March 5, 1999.

According to the above mentioned configuration manual, LANE services configured on a Fore switch are LANE 2.0 compliant. The LNNI v2 specification states that LANE v2 is composed of two parts, LUNI v2, and LNNI v2. Fore's interpretation of LANE 2.0 compliance is with respect to LANE services as seen by LANE clients, not other LANE services. This means that they are only compliant with LUNI v2, but not LNNI v2.

Fore created a Distributed LANE Emulation (DLE) functionality that emulates LNNI v2, but it is quite different. Fore's DLE allows all LANE services (which by the way do not include a SMS) to be addressed via UNI 4.0 "anycast" addressing. This allows a LEC to connect to a LECS and receive an anycast address for all the LES servers in their assigned ELAN. After an LEC gets an anycast address of its ELAN LES server, it sends a

UNI SETUP request to the switch. PNNI routing routes SETUP request to the closest LES server that has that anycast address.

Peer Servers within a Fore DLE ELAN can not be dynamically configured via the LECS and do not have to initially acquire their configuration from the LECS as mandated in the LNNI v2 specification. I is not known if or how Fore peer servers synchronize their LNNI databases or if they perform keepalives with one another or with the LECS. The fact that LE_ARPs' for unknown addresses are forwarded to other DLE servers and to Proxy LECs within a Fore DLE network implies that their LNNI databases are not synchronized. This means that if DLE is implemented in a Fore network, duplicate addressing is possible. Duplicate addressing for reasons already sited in this study is a major security concern.

Fore supports LE_ARP, LE_NARP, and Targetless LE_ARP. Although the use of LE_ARP is a necessity, it and the use of LE_NARP and Targetless LE_ARP represent noted security concerns.

Fore permits a single LES/BUS server implementation, as an alternative to DLE; however, the connectivity process between LECs seems foreign. If a LEC does not know the MAC address of an intended destination it sends an IP-ARP message (this is not the same as a LANE LE_ARP, which resolves ATM to MAC addresses) that is broadcast. If a LEC determines that a broadcast is in reference to an IP address it represents, it establishes (if needed) a virtual circuit to the calling LEC. This connectivity process is different from the typical LUNI connectivity procedure using LE_ARPs. It is assumed that this process may only be used by Proxy LECs, although this has never been verified with Fore.

ELAN access control at a LES implies that LESs can verify LEC join attempts, but the configuration information does not describe any validation procedures for neighbor LES connections, other than synchronization peers must be manually configured within each LES. Even though Fore specifies manual vs. LNNI automatic synchronization peer configuration settings within LESs, rogue ATM devices within a network can register an already valid LES address to validate itself with other LESs in the network; therefore, manual LES configuration should not be construed as a security enhancement.

ELAN access control at a LECS is via ACCEPT and REJECT rules. Fore notes that any rules written should be checked. The ACCEPT and REJECT rules work in two phases, first to check if MAC address or route descriptors rules apply, then for ATM address rules. Fore points out that with some access control rule sets, even though a MAC address may be rejected, if its ATM address isn't rejected then the client is allowed to join. Fore recommend that the rules should be explicit lists of ATM addresses, OR explicit lists of MAC addresses, but not both.

Fore permits "manual," as well as "automatic" signaling assignment. Manual signaling setup involves setting the type of signaling protocol and the version of that protocol used at a port. Automatic signaling setup automatically detects and adjusts to the signaling protocol used by a connected device. In order to prevent rogue client UNI connected workstations from becoming PNNI connected nodes I suggest setting each ports signaling protocol to an appropriate UNI version manually. Note also that when configuring Virtual Path Terminators

(VPT) for Virtual Through Paths (VTP) delete the signaling Channel "0" to prevent PVC connected devices from being able to send signaling messages to your network. This suggestion was noted in conversations with Fore in security related support-related calls.

Fore can do load balance routing. This means that a switch can have several paths to a destination, and randomly assign calls to common destinations over altering paths. It uses shortest path first then (if the paths do not meet call setup traffic descriptors) CTD and CDV metrics are used. The security implications of this are unknown.

Fore has a LECS configuration parameter variable called the "LECS.MAC_Address_Base." This parameter enables the LECS to provide to a LEC, a fourteen-bit pattern to permute its MAC address. This parameter allows for the automatic creation of multiple MAC addresses that are permutations of a workstation base MAC address so that a workstation can have multiple clients or services with unique ATM addresses. The problem with this is there may be other devices or clients in an ELAN that are using the same address. Knowing that UNI will permit duplicate ATM address registrations and that Fore's DLE services may not be synchronized means that intentional or accidental duplicate MAC addresses or duplicate ATM addresses may present a vulnerability if this feature were used.

Fore's Security Features are divided into switch access control, and call setup control. Switch access control security features include: (1) login administrated accounts enabled with various control parameters, (2) login authentication via Kerberos V5, or a third party authentication software from Security Dynamics called "SecureID," and (3) IP address filtering on switch control ports. Call setup control is also provided through NSAP filtering on incoming and/or outgoing UNI ports.

Login administration involves setting "userids"(16 characters), passphrases (512 characters), access methods (either via a front panel serial port on a switch, networked telnet, both, or none), and user privileges (either user type, or administrative type). Administrative type privileges allow a user to change any configuration on a switch. User type privileges are the same as an Administrative type except they prevent any configuration associated with login security (except his or her own passphrase), Kerberos, Secure ID, IP access methods, NSAP filtering, SNMP sets, and the running of debug commands.

Login authentication options include support for the Massachusetts Institute of Technology (MIT) "Kerberos version 5" software, or the Security Dynamics " SecureID" software. The Department of the Navy Naval Information Systems Management Center publishes an "ASSESSED PRODUCTS LIST" (NAVSP P-5239-10); at the time of this writing I was suppprised to learn that the latest version of the ASSESSED PRODUCTS LIST is dated 30 January 1997. No information concerning an assessment of either Kerberos or SecureID was found, except that one software product assessed supported SecureID.

A Kerberos FAQ obtained over the internet from http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html states that Kerberos assumes that "it is running on trusted hosts with an untrusted network.." It also points out that if an attacker compromises a trusted host, than that host can be impersonated, and if a Kerberos

Key Distribution Center (KDC – contains a key database) server is compromised then the entire authenticated user community is compromised. Authentication enhances security, however the choice of an authentication scheme is beyond the scope of this paper. Like Kerberos, SecureID analysis is another authentication scheme and its use is beyond the scope of this paper.

Fore notes [13] that Kerberos and SecurID do not protect against ILMI, SNMP, and remote logins from another switch. Fore also notes that Kerberos doesn't protect against "Element Manager" access to the switch, and SecurID doesn't protect against "ForeView" access to the switch. . Fore recommends the use of IP filtering to allow only authorized SNMP changes, or to disable networked based SNMP "SETs."

IP address filtering limits IP connectivity to a switch control port (which has an IP address) to an individual IP address or groups of IP addresses. It also allows an administrator to set IP filter flags to control the acceptance of strict source, or loose source routed packets, or all packets by the switch control port. Strict source routed packets are packets in which the sender provides a specific path the IP packet must traverse to reach its final destination. Loose source routed packets have only a specific subset of nodes a packet must traverse to reach its final destination. Fore [13] suggests denying any loose source routed, or strict source routed packets because they enable anyone to send a packet anywhere via any route. Fore also provides statistical information functions concerning the denial of failed attempts to access a switch.

NSAP filtering allows NSAP filtering on incomming, and/or outgoing UNI connections associated with specified ports. This should be used to restrict unauthorized NSAP address changes of connected ATM devices. Note however, that if a ATM/LANE connected workstation were infected or compromised in such a way such that its ATM address were changed it would result in a Denial-of-Service attack. Statistical information functions pertaining to accept/reject UNI calls are provided. Fore [13] notes that even with NSAP address filtering, LANE users may still be able to receive data via the BUS from unauthorized sources. Why is this bad? Because NetX clients are Cisco Catalyst 5000s acting as bridges. Bridges receive BPDU topology change and configuration change information via a BUS.

Fore does not require a login from a remote switch. When you login it checks the userid, password, port, and if purchased and setup, Kerberos or Secure ID authentication. Note that if a Secure ID server is down then local authentication only is used. A possible network penetration scenario may involve reducing security procedures in place: A RH spoofs a Secure ID server or disrupts the server somehow such that the switch defaults to local authentication only.

Configuration settings in a Fore switch are made can be stored in a non-volatile configuration database. Non-volatile memory is not lost when the switch looses power.

Some of Fore's older software source code is available on the Internet. This could be analyzed for security weaknesses and modified to do just about anything. Attacks may involve replacing switch software or host software on local or remote systems with a modified version to facilitate other attacks. Modifications to local or remote LEC software could include bypassing LANE Management, and/or Connection Management when changing local host primary or secondary ATM, Mac, and IP address information via direct memory editing. Modifications of switch software may involve the creation of an un-audited "back door" to the switch software. Just about anything is possible if you modify the switch software.

B. NETX CONFIGURATION SUGGESTIONS USING FORE ASX ATM SWITCHES

1. Implement NSAP filtering on every UNI connected port. This prevents UNI signaling messages with augmented unauthorized ATM addresses from being processed by a network. Note that this does not filter secondary ATM addresses registered with LANE LESs. Fore [14] notes that this does not prevent attacks via the BUS. Section VI.A, Capabilities 7, 8, 10, 11, 15 are prevented by this setting.
2. Allow only native ATM addressing and implement multiple PNNI peer groups. Native addressing provides a means to limit the extent of administrative work in configuring NSAP filtering on Fore switches within an ATM/LANE network. It also limits the scope of, or exposes (via network monitoring filters) certain ATM address spoofing attacks that involve changing a local ATM address to that of other devices not connected to a RHs network node. Note that this does not prevent a RH from spoofing other devices that have the same summarized ATM network prefix (connected to the same network node). This suggestion may be considered a second order protective measure for the types of attacks listed in Suggestion 1, if administrative types of errors result in the breakdown of ATM port NSAP address control.
3. In a hierarchical PNNI network disable or limit foreign ATM address advertisements by Peer Group Leaders. This is a third order protective measure in the event that administrative errors occur that result in foreign address registrations. If this situation were to occur then other PNNI Peer Groups would only receive summarized native addresses, or a limited set of authorized foreign addresses. If unauthorized foreign address advertisements were to occur then PNNI could source route calls to RHs that have the longest Most Significant Address Bit Match (MSAB) in PNNI routing tables.
4. Deny strict and loose source routing to control ports. This prevents Ethernet IP address spoofing to a switch control port.
5. Use IP address filtering on control ports. IP address filtering limits control port access to a predefined IP address or set of IP addresses.
6. Configure control port access control with logins, pass-phrases, and permission limits.
7. The use of Fore capable authentication is beyond the scope of this study.
8. Carefully create ELAN access control on a LECS configuration file via Fore recommended procedures. Fore notes a known configuration oversights may result in unexpected permission rights to join ELANs.
9. Use a unique stand-alone ELAN for Fore network management. This suggestion limits the number of devices on the network management ELAN to only trusted controlled network nodes

and the management station itself to reduce the possibility of same-ELAN trusted hosts spoofing the network management station or compromising it somehow.

10. Do not allow UNI or PNNI signaling to network devices outside a locally controlled network security domain. UNI and PNNI signaling provide opportunities (already noted) to compromise a network.
11. Manually configure signaling on all UNI connected ports to the correct version. This suggestion prevents a UNI connected client from reconfiguring itself a PNNI node and ILMI registering itself as a PNNI node. Section VI.A, Item 16 is prevented by this setting.
12. Turn off signaling path 0 on all VPTs associated with PVPs between untrusted ATM connected equipment. This prevents PVC connected ATM devices from being able to signal to devices within your network domain, but it allows untrusted connected ATM equipment to transparently connect through your network and signal to each other.
13. Fore's DLE is security concern if UNI connected Edge Devices are not owned and controlled by the network services provider, or if physical access control measures to ATM devices and interconnected cabling are weak. If the above conditions are not true, and DLE is used, then great care must be taken when designing a network to insure that all risks are known and preventive and defensive measures are taken relating to DLE risks noted in this study.

C. CISCO SYSTEMS EDGE DEVICES AND LANE

The following reflects comments on the Cisco Catalyst 5000 Family Switches and Software documentation publicly available on Cisco's web site.

A Cisco VLAN Membership Policy Server (VMPS) is a server with software that has the capability to allow the dynamic movement of users within a VLAN VMPS domain (consisting of many switches) without having to configure VLAN assignments to port settings on switches manually.

A VMPS domain consists of VMPS servers, clients, and a "rcp" or "tftp" server. The server contains a configuration file that lists MAC address to VLAN name bindings. After a VMPS server is configured and enabled, it downloads a configuration file from a policy server via "tftp" or "rcp." Each VMPS client (that is manually configured with the addresses of the VMPS servers) within a VLAN domain asks a VMPS server to authorize and/or make VLAN assignments to ports when new port MAC addresses are learned. VMPS servers make VLAN assignments based on VLAN/MAC address bindings.

If a VMPS server is configured to be a "secure mode" server, then it can automatically shut down or refuse a port connection on a client when an unauthorized join request is made. I did not find any statements in the Cisco literature that I read that indicates that duplicate MAC addresses on different ports are denied access and VLAN assignment. This may be vulnerability. Port configuration is discussed in detail later in this section.

VMPS servers and clients have their local ports manually configured to be either "static" or "dynamic." Static ports that are set to "not age" deny any MAC/VLAN association changes. Dynamic port assignments allow learned MAC/VLAN assignments to move to other dynamic ports within the VLAN domain.

The "rcp" and "tftp" configuration settings made in VPMS server identify the configuration file server by IP address and configuration file name. Both tftp and rcp protocols are known to be susceptible to attacks.

A VLAN Trunk Protocol (VTP) provides an automatic VLAN creation function within switches identified as being within a VTP domain. VTP domain switches are categorized as being servers, clients, or transparent. VTP servers maintain a synchronized database of VLAN names and identifiers within their domain. When a new VLAN is created on a VTP server, a new VLAN with the same name and ID is created on every other server and client within the domain automatically. This information is exchanged via Inter Switch Link (ISL), 802.1Q, or ATM ELAN trunks. VTP client switches are not permitted to create VLANs. VTP transparent switches do not automatically configure VLANS from a VTP server; they are configured manually. VLAN creation within servers can be made via the Command Line Interface (CLI) or SNMP.

According to Cisco, "VTP minimizes mis-configurations and configuration inconsistencies, preventing duplicate VLAN names, type specifications or security violations." VLAN creation via SNMP is a security concern and should be investigated further.

To minimize the effect of deliberate or accidental broadcast storms, broadcast and multicast suppression filters are provided. Suppression filters can be based on a comparison of actual broadcast bandwidth, multicast traffic bandwidth, or packet rate statistics to predefined threshold settings. This capability could be implemented to reduce the effects of possible Denial-of-Service attacks via intentional broadcast storms.

Access to a switch console port via telnet and SNMP can be controlled with IP address filters. Security traps, and logging can be enabled to report unauthorized console access attempts.

Catalyst switches can be configured to learn and set, or statically set outbound (non-trunking) port protocol filters. Protocol filters are limited to IP, IPX, and group protocols. Port security is further enhanced with static or dynamic per port MAC address assignments, which can remain assigned to a port indefinitely, or aged out. Resulting actions to a MAC address security violation at a port can be set to disable a port, or just to deny unauthorized accesses (Restricted Access).

Although one might want to disable a port in the event of a security violation, I suggest just restricting access. Restricting access prevents Denial-of-Service attacks that could result if a virus or other means causes a MAC address change on a host computer. By setting the port security to "Restricting Access" a hacker has time to scan a port for vulnerabilities, but a network security team has time to try to positively identify the attack and possibly track down the source of such an attack.

I suggest that all ports on all Cisco Catalyst devices be configured with a "Restricted Access" security setting with the quantity of allowed MAC addresses per port set to only the number of known and statically set addresses. All ports not used and configured should be disabled. Port access should be updated by network personnel on a case-by-case basis for host movements, new hosts, or changed host information. A security policy should be written informing users that their network connections are port assigned to their accredited computers Ethernet MAC addresses (unless of course multiple users are connected to a shared hub that is connected to a Catalyst). This policy restricts unauthorized MAC address changes, moves, and adds without network management knowledge.

With policy and procedures in place, a manned 24 hour x 7 day a week network management center could readily respond to device moves adds and changes, as well as to security violations. I have heard time and time again that most attacks are from within This solution will not prevent hosts on a premise network from being compromised, it only confines the scope of damage to a single host and prevents MAC address spoofing. This also suggests separating and limiting the size of trusted host communities, as well as to police intercommunity traffic.

A Cisco Discovery Protocol (CDP) resides on every Cisco device and is enabled globally on every non-trunking port. The CDP enables the discovery of every directly connected Cisco devise by periodically sending information out every port that is CDP enabled. CDP information includes: port-IDs, device IDs, VLAN numbers, IP addresses, the types of devices connected, VTP management domain information, as well as port-IDs of connected devices. CDP messages are not sent through a Cisco device, rather every CDP broadcast ends at directly connected neighbors. Network management applications use CDP information to learn Cisco specific topology information. The CDP can be disabled globally, or on a per-port basis. Ports connecting different security domains should have CDP turned off to minimize network information exposure. Because this information is repeatedly broadcast out every port without request, it can provide unwanted information disclosure to directly connected users.

Cisco has a feature called the Switched Port Analyzer (SPAN) that provides a means to copy traffic to a designated port, or ports for analysis. The source of the copied traffic may be a VLAN, a port, or several ports. SPAN does not actually analyze the traffic, it only copies it to a destination port or ports. The destination ports can include, but are not limited, to a Network Analysis Modules (Cisco plug-in module that extends the RMON capabilities of the switch), Cisco switch probes, ATM analyzers. Multiple SPAN sessions can run at once on a switch, but a limit of 4 VLAN-only SPAN sessions can be associated with trunk ports. If a switch is compromised then this software could allow unauthorized remote data collection and monitoring. Unauthorized SPAN use on a compromised switch, coupled with a compromised host system, would enable for the relaying of real time, or stored filtered information to other location(s) via telephone, or outgoing Ethernet connections.

A Network Analysis Module (NAM) is a Cisco plug-in that extends the Supervisor Engine module RMON capabilities. The NAM can be a destination port for SPAN session. The NAM can be controlled by a network management station via SNMP.

Like Fore Systems, Cisco offers client-server authentication and authorization protocol support. Cisco supports RADIUS and TACACS+. These authentication packages are purchased separately and involve setting up authentication servers. An analysis of the vulnerabilities of these packages is beyond the scope of this paper, however any authentication package should be investigated and used if positive analysis results.

Cisco provides ATM/LANE plug-n modules as well as ATM/LANE/MPOA plug-in modules for its Catalyst switch family. MPOA is not within the scope of this document. The Catalyst 5000 ATM/LANE plug-in modules support LANE v1, and UNI 3.0, and 3.1, but only for IEEE 802.3 ELANs (not 802.5 Token Ring). Therefor LUNI v2 features such as LLC-Multiplexing, qualities of service offerings, or enhanced multicast support as defined in LANE v2 are not implmented.. LNNI v2 features such as distributed synchronized LANE services, Selective Multicast Servers, and standardized configuration, and service interoperations are also not provided. Cisco does offers server redundancy features that enhance LANE v1's limitations via their proprietary Fast Simple Server Redundancy Protocol (FSSRP). The FFSRP only works when all LANE services are configured on Cisco devices, and LECs have FSSRP configured accordingly and enabled.

Cisco's ATM Addressing Structure provides every LEC, LES, and BUS, on every ELAN in a controlled domain with a unique address. Every switch has 1024 preassigned MAC addresses that are allocated to various things, such as VLAN identifiers, ports identifiers, and LANE components such as LECs, LES, BUS, and LECS.

If automatic addressing is used each ATM module is assigned, from the main Supervisor Engine, a list of (16 consecutive MAC addresses). The configuration of an ATM/LANE/VLANs within a switch proceeds as follows:

1. ELAN names are defined and unique Selector Byte values for each ELAN are assigned.
2. If a LES/BUS is being configured on a switch then these are also created with sequential MAC addresses that are associated with an ELAN by a Selector byte assignment.
3. If not already configured, VLANS are created. Each VLAN is assigned a unique MAC address from the main Supervisor Engine MAC address pool.
4. Each ELAN is bound to a single VLAN. When this happens a LEC associated with the ELAN is automatically created.
5. If a LECS is created on the switch then it is assigned a MAC address and its Selector byte is set to 0.

Cisco does offer ways to override the automatic creation of LANE entities' assigned ATM addresses so that addresses can be configured manually. With an ATM address of 20 Bytes and associated labor involved and possible data entry mistakes when configuring a network, there are probably many network administrators who would prefer to have the software automatically address.

With automatic addressing every LEC on a switch has the same Network Prefix, and MAC address. If more than one LEC is configured on a switch, then each LEC must belong to a different ELAN. The ATM addresses of every LEC therefore differ by only one byte. The same is true for every LES on a switch, and every BUS on a switch, i.e., every LES has the same ATM address except the Selector byte, and every BUS has the same ATM address except the Selector byte. All LECs, LESs, BUSs, and LECS configured on a switch have the same Network Prefix, and their MAC addresses are allocated sequentially from the pool of 16 MAC addresses assigned to the ATM module.

Cisco offers distributed LANE services via their proprietary FSSRP. With FFSRP each LES/BUS associated with an ELAN is configured into the LECS database. LECs have connections to every LES/BUS pair in their ELAN; however, they only use one active pair with redundant LES/BUSs connections idle waiting in hot-standby for the LECs to use in the event of active LES/BUS connection failures.

D. NETX CONFIGURATION SUGGESTIONS USING CISCO CATALYST EDGE DEVICES.

1. Allow only "static" port MAC address assignments. This prevents Ethernet connected hosts from spoofing other hosts via MAC address changes.
2. Do not allow port addresses to "age." This prevents possible MAC address spoofing.
3. Use "Restricted Access" port security. This prevents Denial-of-Service types of attacks.
4. The quantity of allowed address assignments for each port should be restricted to only the number of devices that are registered and approved to connect to a port. This prevents Ethernet connected hosts from spoofing other hosts via MAC address changes.
5. Avoid using Cisco's default ATM addressing for ATM/LANE extended ELANS/VLANs. This suggestion complicates an ATM addressing structure to throw-off hack attempts using inferred Cisco specific ATM addressing procedures.
6. If VMPS is implemented, only connect or enable a VMPS (rcp or tftp) server when needed. The "rcp" and "tftp" protocols are not secure.
7. Cisco recommends using VTP to minimize configuration errors.
8. Limit CDP broadcasts to only those ports connected to other Cisco devices (hardware/software dependent suggestion).
9. Use SPAN on every Catalyst device. This is a standard way to not only allow troubleshooting of network problems, but it also allows network monitoring devices to detect security related traffic anomalies.
10. Enable broadcast and multicast suppression filters where appropriate.
11. Cisco recommends IP address filters, security traps, and logging.
12. Use console logins, passwords, and privileges settings.
13. Outbound port protocol filters may be used where appropriate. One setting may group protocols from causing Denial-of-Service attacks.

E. OTHER NETX SECURITY SUGGESTIONS

1. Every ATM/LANE device must be within a controlled security domain and physically secured to prevent access from unauthorized personnel. Section VI.A items 1, 2, 3, 4, 5, 6, 9, and 12 are prevented by this suggestion.

2. Physical media connecting ATM/LANE devices must be installed with some level of access control that is appropriate for the highest classification level of data that it supports and the lowest classification level of the areas through which it passes. Access control may include government approved encryption and authentication algorithms. Even with approved government encryption, note that with the increased speed of computer processing each year the lifetime of encrypted data secrecy is reduced. Therefore the classification lifetime of the data that is encrypted must be considered. This however is a topic which should be addressed by the government for further study and recommendations. . Section VI.A items 13, and 14 are prevented by this suggestion.
3. Physical cable plant designs of connected ATM/LANE devices should provide path redundancy appropriate to the criticality of data delivery expected by the clients it supports. This is suggested to prevent Denial-of-Service types of attacks or to prevent other types of attacks due to a Denial-of-Service.
4. Network Management Stations (NMS) should only be configured with NMS software, virus scanning software, and other software specific to network management and security. All office automation type packages such as mail, or web browser packages used for network management should be used with caution. An example is to only allow outgoing mail to alert administrators of problems, or web browsers should only be used to connect to trusted sites for network management and security related software downloads.

THIS PAGE IS INTENTIONALLY LEFT BLANK

IX. CONCLUSION

The purpose of this study is to provide: (1) a description of the operation of standardized ATM and LANE related protocols and an investigation of related security issues, (2) an overview and investigation of security issues associated with integrating a Fore Systems Inc., LANE based ATM backbone network into an accredited Cisco Systems Inc., based Ethernet Virtual LAN network, and (3) security related suggestions for network design and configuration specifically with regard to LANE and Fore Systems Inc, ATM backbone switches interconnecting Cisco Catalyst based edge devices.

Chapters II, III, IV, and V describe the operation of the TCP/IP protocol suite, LANE, and ATM related protocols including UNI and PNNI, as well as an example of data transport. Chapter VI and VI.B provide high level negative security related protocol capability summaries. Chapter VII provides supporting protocol and plane specific security point details as well as additional security related discussions. Chapter VIII provides an overview of Fore and Cisco network operations and security points as well as specific suggestions for securing a model network called "NetX."

This study shows that there are a number of possible protocol specific negative security-related capabilities associated with UNI based connections to ATM networks. The risks imposed by these capabilities are significantly reduced if suggested network device configuration settings and network implementation recommendations made in this study are used.

This study qualitatively shows that bit and burst errors along LANE based ATM communications paths do not increase the probability of corrupted frames from being delivered to, received, accepted and processed by TCP/IP based applications at incorrect Ethernet destinations.

Protocol and Vendor related security points as well as transmission error outcomes are the result of an analysis of the relevant standards, and publicly available vendor product literature. This study does not reflect actual network implementations, nor expose verified vulnerabilities, nor means to exploit these vulnerabilities.

Configuration options offered by Fore and Cisco are provided for a purpose that is often misunderstood. This thesis should clarify the intent and importance of those configuration options noted by the vendors, as well as provide additional suggestions to better secure a network.

It is hoped that this seminal document will assist in the development of standard security-driven implementation and operation procedures associated with ATM/LANE based networks, as well as to inform those employees required to prepare, and review, associated network Risk Assessments.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1 M. Peyravian and I.D. Tarmanh, "Asynchronous Transfer Mode Security," pp. 34-39, IEEE Network, May/June 1997.

2 The ATM Forum, "ATM Security Specification, Version 1.0," af-sec-0100.000, February 1999.

3 The ATM Forum, "ATM Security Framework, Version 1.0," af-sec-0096.000, February 1998

4 The ATM Forum, "LAN Emulation over ATM, Version 2 – LUNI Specification," af-lane-0084.000, July 1997.

5 The ATM Forum, "LAN Emulation over ATM, Version 2 – LNNI Specification," af-lane-0112.000, February, 1999.

6 Luciani, J., et al, "RFC 2334: Server Cache Synchronization Protocol (SCSP)," April 1998.

7 The ATM Forum, "ATM User-Network Interface Specification, Version 3.1," ATM Forum Specification, September 1994.

8 Black, U., "ATM Volume I – Foundation for Broadband Networks," Prentice Hall, 1995.

9 American National Standards Institute T1.105, "Synchronous Optical Network (SONET) Basic Description including Muliplex Structure, Rates, and Formats," American National Standards Institute, Inc., October 1995.

10 Doshi, B., et al, "Scramblers for PPP over SONET/SDH: Considerations and Analysis," contributions to T1 standards Project T1X1.5/97-129, Lucent Technologies, December 1997.

11 American National Standards Institute T1.646, "Broadband ISDN - Physical Layer Specifications for User-Network Interfaces Including DS1/ATM," American National Standards Institute, Inc., May 1995.

12 ITU-T, Draft Recommendation I.35B, "Broadband ISDN Performance," WP XVIII/6 (part 2, draft recommendations), TD 15 (XVIII)].

13 Fore Systems, "ATM Switch Network Configuration Manual," MANU0148-06 Revision A, Fore Systems, Warrendale, PA, March, 1999.

14 Fore Systems, "AMI Configuration Commands Reference Manual – Part 2," MANU0265-05 Revision A, Fore Systems, Warrendale, PA, March, 1999.

THIS PAGE IS INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
8725 John J. Kingman Rd., STE 0944
Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Rd.
Monterey, CA 93943-5101

3. Chairman, Code EC 1
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, CA 93943-5121

6. Curricular Officer, Code 34 1
Engineering and Technology
Naval Postgraduate School
Monterey, CA 93943-5109

7. Professor John McEachen, Code EC/Mj 1
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, CA 93943-5121

8. Professor Murali Tummala, Code EC/Tu 1
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, CA 93943-5121

9. Ms. Rosemary Wenchel 1
Naval Information Warfare Analysis Center
1800 Savage Road
Ft. Meade, MD 20755

10. CAPT. John O'Dwyer 1
Commanding Officer
Naval Information Warfare Activity
9800 Savage Road
Ft. Meade, MD 20755

11. Laboratory for Telecommunication Science.. 1
9800 Savage Road
Ft. Meade, MD 20755
ATTN: R56

12. National Security Agency.. 1
9800 Savage Road
Ft. Meade, MD 20755
ATTN: R22

INITIAL DISTRIBUTION LIST (CONTINUED)

No. Copies

13. National Security Agency.. 1
9800 Savage Road
Ft. Meade, MD 20755
ATTN: C42

14. National Security Agency.. 1
9800 Savage Road
Ft. Meade, MD 20755
ATTN: K72

15. Naval Surface Warfare Center 1
Dahlgren Division
Code T11, Bldg 1200
17320 Dahlgren Rd
Dahlgren, VA 22448-5100
ATTN: John Kirwin